Dipartimento di Informatica, Bioingegneria,
Robotica ed Ingegneria dei Sistemi

Data-Driven and Hybrid Methods for Naval Applications

by

Francesca Cipollini

Theses Series **DIBRIS-TH-2019-01**

DIBRIS, Università di Genova
Via Opera Pia, 13 16145 Genova, Italy http://www.dibris.unige.it/

Università degli Studi di Genova

Dipartimento di Informatica, Bioingegneria,

Robotica ed Ingegneria dei Sistemi

Ph.D. Thesis in Computer Science and Systems Engineering
Systems Engineering Curriculum

Data-Driven and Hybrid Methods for Naval Applications

by

Francesca Cipollini

October, 2019

Dottorato di Ricerca in Informatica ed Ingegneria dei Sistemi
Indirizzo Ingegneria dei Sistemi
Dipartimento di Informatica, Bioingegneria, Robotica ed Ingegneria dei Sistemi
Università degli Studi di Genova

DIBRIS, Univ. di Genova
Via Opera Pia, 13
I-16145 Genova, Italy
http://www.dibris.unige.it/

Ph.D. Thesis in Computer Science and Systems Engineering
Systems Engineering Curriculum
(S.S.D. INF/01)

Submitted by Francesca Cipollini
DIBRIS, Univ. di Genova
francesca.cipollini@edu.unige.it

Date of submission: October 2019

Title: Data-Driven and Hybrid Methods for Naval Applications

Advisor: Prof. Luca Oneto
Dipartimento di Informatica, Bioingegneria, Robotica ed Ingegneria dei Sistemi
Università di Genova
luca.oneto@unige.it

Ext. Reviewers: Prof. Joaquín Luque
Universidad de Sevilla
jluque@us.es,
Prof. Attila Kertesz-Farkas
Higher School of Economics
kfattila@yandex.com
Prof. Andrea Coraddu
University of Strathclyde
andrea.coraddu@strath.ac.uk

Abstract

The goal of this PhD thesis is to study, design and develop data analysis methods for naval applications.

Data analysis is improving our ways to understand complex phenomena by profitably taking advantage of the information laying behind a collection of data. In fact, by adopting algorithms coming from the world of statistics and machine learning it is possible to extract valuable information, without requiring specific domain knowledge of the system generating the data.

The application of such methods to marine contexts opens new research scenarios, since typical naval problems can now be solved with higher accuracy rates with respect to more classical techniques, based on the physical equations governing the naval system. During this study, some major naval problems have been addressed adopting state-of-the-art and novel data analysis techniques: condition-based maintenance, consisting in assets monitoring, maintenance planning, and real-time anomaly detection; energy and consumption monitoring, in order to reduce vessel consumption and gas emissions; system safety for maneuvering control and collision avoidance; components design, in order to detect possible defects at design stage.

A review of the state-of-the-art of data analysis and machine learning techniques together with the preliminary results of the application of such methods to the aforementioned problems show a growing interest in these research topics and that effective data-driven solutions can be applied to the naval context. Moreover, for some applications, data-driven models have been used in conjunction with domain-dependent methods, modelling physical phenomena, in order to exploit both mechanistic knowledge of the system and available measurements. These hybrid methods are proved to provide more accurate and interpretable results with respect to both the pure physical or data-driven approaches taken singularly, thus showing that in the naval context it is possible to offer new valuable methodologies by either providing novel statistical methods or improving the state-of-the-art ones.

Acknowledgements

Firstly, I express my sincere gratitude to my supervisor Prof. Luca Oneto, for the continuous support of my study and related research, for his patience, motivation, and trust towards my work and interests.

I thank all the AizoOn colleagues, for their support and sympathy during these three years of PhD. In particular, I would like to thank Ing. Francesco Volpi for his constant help and trust.

I also thank Prof. Andrea Coraddu, for his dedication and support during all the projects we developed; without his experience in Naval Applications, obtaining the results presented in this thesis would have been extremely hard.

Their guidance helped me in all the time of research and writing of this thesis.

Moreover, I would like to thank all the SmartLab components, which supported me during my PhD period.

My sincere thanks finally goes to my family and Giorgio, who always supported me in my study.

Table of Contents

List of Figures

List of Tables

List of Abbreviations

AAW	Anti-Aircraft Warfare
AMR	Average Misclassifications Rate
ASW	Anti-Submarine Warfare
BEM	Boundary Element Methods
BM	Bayesian Method
BTS	Bootstrap
CBM	Condition-Based Maintenance
CFD	Computational Fluid Dynamics
CM	Corrective Maintenance
COLREG	International Regulations for Preventing Collisions at Sea
DA	Data Analysis
DELM	Deep Extreme Learning Machine
DDM	Data-Driven Model
DG	Diesel Generator
DNN	Deep Neural Network
ELM	Extreme Learning Machine
EM	Ensemble Method
EPM	Electric Propulsion Motor
ERM	Empirical Risk Minimisation
FM	Feature Mapping
FS	Feature Selection

FR	Feature Ranking
GEP	General-Purpose
GB	Gearbox
GT	Gas Turbine
GTC	Gas Turbine Compressor
HDD	Heading Deviation Direction
HLL	Hull
HM	Hybrid Model
HR	Head Reach
HSC	High-Speed Craft
IM	Induction Motors
ITL	Independent Task Learning
ITTC	International Towing Tank Conference
LD	Lateral Deviations
LDD	Lateral Deviation Direction
LM	Lazy Method
KCV	K-Fold Cross Validation
KM	Kernel Method
KNN	K-Nearest Neighbours
KRLS	Kernelized Regularized Least Squares
MAE	Mean Absolute Error
MAPE	Mean Absolute Percentage Error
ML	Machine Learning
MSE	Mean Squared Error
MST	Model Scale Test
MTL	Multi Task Learning
MTU	Motor-Transmission-User
NMSE	Normalized Mean Square Error

NN	Neural Network
NPS	Naval Propulsion System
OCSVM	One-Class Support Vector Machines
PCA	Principal Component Analysis
PM	Physical Model
PPMCC	Pearson Product Moment Correlation Coefficient
PRM	Predetermined Maintenance
PRP	Propeller
REB	Rolling-Element Bearing
REP	Relative Error Percentage
RNL	Radiated Noise Level
RF	Random Forest
RRE	Random Rotation Ensemble
SELM	Shallow Extreme Learning Machine
SNN	Shallow Neural Network
STL	Shared Task Learning
SVM	Support Vector Machines
SVR	Support Vector Regression
TFM	Time for Full Maneuver
TR	Track Reach
URN	Underwater Radiated Noise

List of Machine Learning Symbols

$\mathcal{X}$	Input space $\subseteq \mathbb{R}^d$
$\mathcal{Y}$	Output space
$\boldsymbol{x}_i$	Input vector belonging to $\mathcal{X} \subseteq \mathbb{R}^d$
y_i	Output point belonging to $\mathcal{Y} \subseteq \mathbb{R}$
μ	Rule associating $\mathcal{X} \rightarrow \mathcal{Y}$
$\mathcal{H}$	Set of hyperparameters
$\mathscr{A}_{\mathcal{H}}$	Algorithm adopting the set of hyperparameters $\mathcal{H}$
$\mathcal{F}_{\mathcal{H}}$	Set of models proposed by $\mathscr{A}_{\mathcal{H}}$
f	Model in $\mathcal{F}_{\mathcal{H}}$
$\mathcal{D}_n$	Dataset composed by $\{(\boldsymbol{x}_i, y_i)\}$ elements
$\mathcal{L}^r_{n_l}$	Training set composed by n_l samples of $\mathcal{D}_n$
$\mathcal{V}^r_{n_v}$	Validation Set composed by n_v samples of $\mathcal{D}_n$
$\ell(f(\boldsymbol{x}), y)$	Loss function
$L(f)$	Generalization Error
$\widehat{L}(f)$	Empirical Error
$\boldsymbol{\varphi}$	Feature Mapping from $\mathcal{X} \rightarrow \Psi$

Chapter 1

Introduction

1.1 Background

One of the main objectives of shipwrights companies has traditionally been to improve the technological quality of their products, e.g. by designing more efficient hull shapes and propeller geometries, studying innovative propulsion systems, and reducing the overall production costs [Car12]. As a result, many shipping industries have based their competitiveness and strategic decisions on the ability to improve these technologies, by leveraging on empirical experience and on very complex Physical Models (PMs) built upon a priory knowledge of the specific problems [COBA17]. This approach, despite relying on the physical equations representing the problem to be solved, clearly requires a significant amount of time and experience to be adopted, still being subject to inaccuracies [MCO+19].

Contemporary, during the last decades, vessels have been equipped with many on board sensors for different purposes, such as automation, quality check, monitoring, and logging [PJW12]. The result of this process is the availability of a huge amount of historical and real-time data thanks to the newly developed digital technologies. In fact, sensors are nowadays embedded in most of the main new-built ship components and allow the ship owners and builders to monitor virtually every aspect of the vessel operations and use. Recently, many shipbuilding companies have realised that these data, despite their management costs, can be considered as an excellent opportunity to improve their business enacting an "Actionable Initiative" [BL12]. These data, constantly created by both humans and machines, are relevant sources of information since they can be used for many business-related purposes, by looking for interesting trends in their stream in time. Recent trends in the shipping industry show that the data growth will continue, and therefore there is a need to provide ship owners and operators with efficient solutions able to support them during the decision processes based on ship data statistics and evidence [MMO18]. Actually, this exponentially growing heterogeneous amount of data, extracted from different sources,

is denominated as Big Data and it is characterised by the so called "Gartner's 5 V" [DGDLM13]:

- Velocity: the speed with which data are acquired and change;
- Variety: the many forms and standards on which the data appear;
- Volume: the amount of data to store and analyse;
- Veracity: the messiness or trustworthiness of the data;
- Value: the relevance of the information stored for the business goal.

In the naval context, these properties are outstanding since ships are extensive scale complex engineering systems, composed of many subsystems and components, whose interactions are fundamental to understand the overall system performance [GRRE12]. As a result, in the latest years, these shipbuilding companies have been working in order to exploit the availability of this ever-growing amount of data coming from multiple sensors present on-board.

The rise in data volumes, computational power, and connectivity have brought to the development of different enabling technologies, providing greater reliability and lower costs with respect to classic naval techniques. In particular, with the term Data Analysis (DA) is usually indicated the ability of producing knowledge from data, adopting methods and tools directly coming from the world of statistics and mathematics, and applying them to data with the aim of discovery. Machine Learning (ML), Data Mining, Computational Intelligence, Operational Research and Optimisation are also part of the DA world [Giu05]. Nevertheless, in order to execute this kind of analysis, large storage of data is requested, and the more information are tracked and stored, the more the technology needs to find clever ways to analyse great amount of data reducing the computational time necessary for this operation [MW15].

Many Data-Driven Models (DDMs) available nowadays come from the worlds of statistics and ML, and they are used in order to analyse the data collected from the concerned system. In particular, ML algorithms deal with the problem of extracting information or patterns of behaviour from data, without any prior knowledge of the system which has generated them [Vap98]. In this way, the fundamental concepts adopted to build these models can be applied to an infinite amount of different problems, with the help of proper statistical knowledge. By taking advantage of these DDMs based on the collected data, significant performance improvement and cost savings can be achieved. For example, historical data of the system can be exploited by ML methods in order to create new services or improve the quality of the products sold by the companies or, data analysts can adopt them to build DDMs, which can rapidly extract actionable information in order to proficiently improve the vessel's performances, or identify malfunctions, or maintain it. This leads to a more accurate monitoring and control of a complex physical system such as a ship.

As a consequence, the goal of this thesis is to study, design and develop DA methods for different naval applications, by levering on large collection of sensor data. In the following section, the fields of applications of the proposed methods are presented.

1.2 Applications

As discussed in Section 1.1, the monitoring of a ship equipment can be achieved by exploiting heterogeneous on-board sensors. Data collected with this complex network of sensors can be used to develop effective DDMs, enabling diagnosis and prognosis of the system's components and their potential future failures. Since that DDMs are based on statistical inference on the collected data, this kind of models can be mapped to any sort of problem or application where a collection of information is present.

During this study, many applications of the DA methods and technologies are proposed, to target different tasks. In particular, DA is applied to improve the following naval applications:

- Condition-Based Maintenance, consisting in assets monitoring, maintenance planning, and real-time anomaly detection;
- Energy and Consumption Monitoring, in order to reduce vessel consumption and gas emissions;
- System Safety for maneuvering control and collision avoidance;
- Components Design, in order to detect in advance possible defects.

Firstly, much interest converges in the Condition-Based Maintenance (CBM) analysis on different sets of data. A correct maintenance program ensures that the studied system works as it was designed, with the desired level performances, without impacting the service [MCI84]. Maintenance policies can be divided into two main categories [BB09, AES10]: Corrective (CM), and Preventive (PM). CM has been for many years the only way of performing maintenance, by replacing a component only after its breakdown, thus compromising the overall system availability and causing exceptional costs and loss in incomes [KH07]. In PM, instead, a component is replaced when it reaches the end of its life cycle before a possible breakdown. One of the traditional ways to perform PM is to predetermine a conservative average estimation of the component time-to-live adopting the experience gained with all the components belonging to a specific class [SA11]. Similarly to CM, this particular type of PM, usually called Predetermined Maintenance (PRM), can bring unnecessary costs, if the replaced component could have been used more than originally forecast. Moreover, PRM does not guarantee to limit the number of faults in a fleet, since a breakdown could still happen before the replacement takes place. In this case, there is a trade-off between the number of breakdowns and the lifetime estimation of the components, which is not easy to reach since the actual components'usage can be very different. Nevertheless, CBM can be considered as another way of performing PM, which aims at reducing both the costs of CM and PRM by relying on the exact decay state of each component and then by efficiently planning its maintenance [MSK95, ISO04]. Note that, condition monitoring and failure prediction are two different concepts which are somehow strictly related. In fact, a failure of a component is predictable only if it is preceded by a decay in its performance or in the performance of some related component [GYO10, JBW$^+$01]. Considering the estimated state of decay, it is possible to schedule each component's replacement before accidents may occur,

maximising its life cycle, according to the time required for each maintenance. As a result, the additional costs of maintenance can be replaced with the lower ones of equipping the system with sensors and by collecting, storing, and analysing these data for the purpose of creating effective predictive DDMs.

Another topic which is considered during these studies is the environmental impact of shipping activities. As a matter of fact, shipping is responsible for approximately the 90% of world trade leading to significant impacts on the environment. Subsequently, a crucial issue for the maritime industry is to develop technologies able to increase the ship efficiency, by reducing fuel consumption and unnecessary maintenance operations. As awareness of climate change increases, new research results keep confirming that there is a need of fast, and strong action [IPC18]. International maritime transport, while representing approximately 90% of global trade and the backbone of global economy, contributes to approximately 2.7% of the global anthropogenic carbon dioxide (CO_2) emissions [SJA+14]. While this might appear a limited contribution, if current trends are not changed shipping will become one of the largest shares of global emissions [AB12], since as of today, ships are still almost entirely powered by fossil fuels. While alternative fuels have shown to be promising [GWT+18], there is a need for ship energy systems to become more energy efficient [LMJR17]. As a result, sustainable shipping is recognised as one of the biggest challenges of the 21st century[1], both for its contribution to CO_2 emissions and to other pollutants [SvdLGP18]. In recent years, however, the International Maritime Organisation (IMO) has officially adopted an initial strategy aiming at reducing Greenhouse Gases (GHG) emissions from shipping by 50%, compared to 2008 levels, by 2050, and to work towards phasing out them entirely by the end of the century [MEP18]. As a consequence, developing new technologies able to both improve the design of the ships and to maintain their efficiency becomes a crucial issue [DKTA13].

A further topic which is studied during this thesis is the vessel safety system concerning maneuvering control and collision avoidance. In fact, shipping is one of the most safety critical industries [Bal06], and the insurance of safe navigation is one of the main responsibilities of the International Regulations for Preventing Collisions at Sea (COLREG). Globalisation has led to increased marine traffic in the recent decades and therefore has made evident the importance of safe navigation. Ship collisions with other moving or stationary objects remain a most relevant problem during navigation. In fact, specific manual maneuvering is requested to fully control the ship while entering or departing from a harbour, or crossing canals and traffic zones, since in these conditions many exogenous factors such as the effects of the wind and currents are not directly controllable nor predictable [HA13]. Ship collision, not only causes structural damages to the hull, but most importantly leads to loss of lives and property, and can cause irreversible marine pollution [Int00, Int94]. As a result vessel manuevering safety has been subject to great interest in the last years [SHHB12, DRK14].

The last topic that is discussed in this work is the optimisation of a ship components design.

[1] http://www.ssi2040.org

In the years, as ships are complex structures, computers have become a constant in the design bets-practices, in order to model each ship component in an efficient and satisfying way. For example, Computational Fluid Dynamics (CFD) models [BB10b, BB10a] can predict the ship behaviour and efficiency only knowing the ship main components geometry and operating conditions. Nevertheless, the CFD high computational requirements limit their use to the field of research, being it impractical in a normal ship design loop. Another potential approach is to build a physical prototype of the ship main components and test them. Anyway, this approach is affected by some disadvantages too: firstly, it is very expensive and time-consuming to both build a prototype of a ship component and test it in a dedicated structure; secondly, the prototype could be affected by some scale which need to be considered and accurately transposed in the full scale. These characteristics make MSTs to be impractical in the early stage of the components design. Nevertheless, both CFD and MST could allow to collect a large amount of data suitable to build a statistical model able to mimic their behaviour, with less requirement of time and resources. This data-driven model could be used in order to explore some combinations of all the parameters of a component, without the necessity to run many costly CFD simulation or to build several prototypes of the component to get the best parameter combination.

Nevertheless, there are many problems which can affect a DA project in the naval context. First of all, DA is a computational intensive task, requiring high processing power and vast storage areas; however, if in the past a single computer could well perform a small task, nowadays, it is no longer suitable due to its limited main memory. Parallel computing and collective mining are the most promising solutions when analysis on large amount of data is requested, but these technologies bring a new model of projecting applications and software, considering the underlying physical architecture of the machines'cluster. Parallelization can thus make systems meet the scalability and performance requirements that Big Data application demand. Nevertheless, the benefits of parallelization come at the costs of re-writing the ML algorithm in a parallel-wise way, and maintaining or purchasing a cluster of machines.

Secondly, as for every specific industrial problem, a field expert is required in order to properly define the interesting and correct data and information to be considered to solve a specific task. In fact, usually data tend to be affected by noise and spurious observations, and require cleaning and preprocessing operations, which become easier with the guidance of a technical expert. In addiction, the intervention of an expert should be required when labelling the data; nevertheless, this operation might not be feasible for cases where large amount of data need to be labelled. A possible but complex solution would be to approximate the target label from data themselves, adopting unsupervised techniques, which can lead to higher inaccuracies.

Finally, since most of DA techniques can be applied on a collection of data, independently on the system from which they come from, as ML models are context-independent, they provide a black-box solution, which give little or no hints on the phenomena which are governing the data. In some cases, this characteristic is a drawback if causal consequences are investigated (e.g. ship faults) or relationship between variables and sensors are demanded. To solve this issue, a new

family of methods, called grey-box models or Hybrid Models (HMs) was born in the latest years, which allows combining more traditional physical-based models with data-driven ones. In this context, the cooperation with a field expert is even more crucial, in order to create a model able both to satisfy accuracy bounds, generalising on the data collection, and to give a theoretical explanation of the physical phenomena which are involved.

In this thesis, where the application made it possible, HMs have been adopted and compared with state-of-the-art physical and data-driven techniques. In particular, the novelty of this thesis stands in a particular type of HM which is here proposed in the context of a ship components design optimisation.

1.3 Structure of the Thesis

The thesis structure is reported in the following.

Chapter 2 contains a full review of the main ML concepts and algorithms in order to extract information from data. In Section 2.1 the most used error measures are reported, for both regression and classification tasks. In Section 2.2 the Model Selection procedure to select the most suitable ML model dependently to the task to pursue is explained. In Section 2.3 the most used Feature Mappings are reported, along with the most used Kernels. In Section 2.4 the most common Supervised ML algorithm are reported, respectively divided in their ML algorithm families, such as Neural Network (2.4.1), Kernel Methods (2.4.2), and Ensemble Methods (2.4.3). Section 2.5 reports some Semi-Unsupervised Anomaly Detection algorithms which allow to avoid the limitation of many Supervised ML algorithms. Finally, Section 2.6 reports some techniques which can be adopted in order to create HMs, able to combine more traditional physical-based models with data-driven ones.

In Chapter 3 some applications are presented which make use of the tools and algorithms presented in Chapter 2 in order to succeed in their specific purposes.

In Section 3.1 the CBM of a naval propulsion system is proposed. During this project, ML algorithms are applied to accurately predict a vessel propulsion system components decay, since, in most cases, the decay state of the components cannot be tracked with a sensor, and a ML model able to predict it based on other sensors available is requested. Furthermore, since label sensor data can be quite expensive and in some cases unfeasible, an unsupervised approach is tempted to monitor the ship conditions, in order to minimise the feedback of the operators in labelling the sensor data. It is proved that it is possible to treat a CBM problem in an unsupervised fashion, with results close to the ones obtained with supervised techniques.

In Section 3.2 an unintrusive monitoring of induction motors bearings adopting stator currents is presented. During this project, the state of a set of induction motors bearings is studied, in order to monitor their performance decay during operations for maintenance purposes [COC+18c]. A

monitoring tool is proposed, which leverages on stator currents signals processed with a Deep Learning architecture, in order to both visualise and predict the induction motors bearings status. Differently from the state-of-the-art approaches which exploit vibration signals, collected by easily damageable and intrusive vibration probes, the stator currents signals are commonly available, or easily and unintrusively collectable. Moreover, a Deep Learning architecture is exploited, able to extract from the stator current signal a compact and expressive representation of the bearings state, ultimately providing a bearing fault detection system.

Section 3.3 reports a data-driven digital twin of a ship for estimating the speed loss due to the marine fouling [COB$^+$ed]. Fouling affects the efficiency and the performances of a ship by increasing its hull and propeller water resistance thus causing an increase in power consumption to keep a constant vessel speed. In this context, a data-driven Digital Twin of the ship is proposed, leveraging on the large amount of information collected from the on-board sensors, and used it for the purpose of estimating the speed loss due to marine fouling. In particular, the speed loss is estimated by computing the error between the Digital Twin prediction and the real value of the speed, in order to determine the best time for scheduling a dry-docking for cleaning operations. The proposed model is compared with the ISO 19030, the de-facto standard for this task, by adopting real-world data coming from two Handymax chemical/product tankers, to assess the effectiveness of the proposal.

In Section 3.4 a data-driven solution for safety and collision avoidance in crash stop maneuvering performance is presented. In this project, the monitoring of the crash stop manoeuvring performance of a set of vessels is targeted, for the preliminary assessment of safety requirements imposed by the classification societies [OCC$^+$18]. The focus is on predicting accurately and with minimal computational effort the crash stop characteristics of a vessel in the design stage. Moreover, different classes of possible vessels were considered and applied the proposed method to different subsets of them, in order to extrapolate on different kind of sets and test the generalisation performances of the algorithm on each possible combinations of the vessels sets. In particular, a new data-driven method was proposed, based on the Random Forests learning algorithm, for predicting the crash stop manoeuvring performance.

Section 3.5 presents a HM for the model scale cavitation noise spectra prediction, adopting both more classical naval approaches and data-driven ones. In detail, during this project, the problem of predicting the underwater radiated noise produced by a ship propeller during cavitation is targeted. The adoption of a HM was proposed, combining physical knowledge on the noise and data-driven technique in order to predict a parametrization of the cavitation noise spectra, specifically developed during this project to best represent its main frequency-domain characteristics. The process is validated with the test data collected at the cavitation tunnel at the University of Genoa and proved to be more effective than the tradition PM and the purely machine-learning based approaches.

In Chapter 4 conclusions are drawn.

Chapter 2

State of the Art

In the latest years, the concept of Data Science has been under the spotlight for many reasons. Firstly, the technological advances allowed the storage of large collection of data, which a few years ago would have been intolerable to bear for both costs and space reasons; secondly the computation capabilities of computers increased so much that some particular resource-demanding algorithms can be performed without a broad time horizon for completion. Finally, many problems which in the past were insolvable, started to be approached by data scientists, such as medical diagnosis [MS10], financial analysis [SLK05], image recognition [LB$^+$95], and product recommendation [PB07].

[Hay98] writes that "the aim of data science is to reveal the features or the hidden structure of complicated natural, human and social phenomena with data from a different point of view from the established or traditional theory and method". In other words, DA can be adopted both in alternative or in conjunction with more traditional techniques in all those cases were a collection of information is present [Vap98].

ML can be considered as a leading segment of the DA techniques, since it involves the creation of artificial models which are able to perform specific tasks without being explicitly programmed to do so. In fact, a typical ML framework considers an input space $\mathcal{X} \subseteq \mathbb{R}^d$ and an output space $\mathcal{Y}$, where ML techniques aim at estimating the unknown rule $\mu : \mathcal{X} \to \mathcal{Y}$ which associates an element $y \in \mathcal{Y}$ to an element $\boldsymbol{x} \in \mathcal{X}$. A ML technique estimates μ through a learning algorithm $\mathscr{A}_{\mathcal{H}} : \mathcal{D}_n \times \mathcal{F} \to f$, characterised by its set of hyperparameters $\mathcal{H}$, into a function $f : \mathcal{X} \to \mathcal{Y}$.

Independently from the adopted technique, any model f requires some data in order to be tuned (or learned) on the problem specificity and to be validated (or tested) on a real-world scenario. For these purposes, two separate sets of data $\mathcal{D}_n = \{(\boldsymbol{x}_1, y_1), \cdots, (\boldsymbol{x}_n, y_n)\}$ and $\mathcal{T}_m = \{(\boldsymbol{x}_1^t, y_1^t), \cdots, (\boldsymbol{x}_m^t, y_m^t)\}$ sampled i.i.d from μ need to be exploited, to respectively tune h and evaluate its performances. It is important to note that $\mathcal{T}_m$ is needed since the error that h would commit over $\mathcal{D}_n$ would be too optimistically biased since $\mathcal{D}_n$ has been used to tune h.

ML techniques, can be grouped according to many affinity principles, in particular all algorithms can be divided into two families:

- Supervised Methods, when both $\boldsymbol{x}_i$ and y_i with $i \in \{1, \cdots, n\}$ are available for n dimension of the dataset. To this category belong Regression and Classification techniques. The first ones are characterised by $\boldsymbol{y} \in \mathbb{R}$, while the second ones are characterised by an output space composed of a finite set of c possibilities, $\mathcal{Y} \in \{C_1, \cdots, C_c\}$;
- Unsupervised Methods, when only $\boldsymbol{x}_i \in \{1, \cdots, n\}$ is available and the association to y_i is not explicitly known. In this case, it has to be assumed that "similar" $\boldsymbol{x}_i$ are associated with "similar" y_i where the concept of similarity is something that needs to be defined based on μ.

It has to be noted that mainly supervised techniques are discussed in this work.

The purpose of any supervised learning procedure is to select the best h such that the expected error

$$L(f) = \mathbb{E}_\mu \ell(f(\boldsymbol{x}), y), \tag{2.1}$$

is minimum, where $\ell(f(\boldsymbol{x}), y)$ is the Loss function measuring how much the prediction $f(\boldsymbol{x})$ is close to y. It has to be noted that since μ is unknown, $L(f)$ is unknown too, and thus an estimation of the real value is needed, and defined as the empirical error

$$\hat{L}^{\mathcal{D}_n}(f) = \frac{1}{n} \sum_{(\boldsymbol{x},y)\in\mathcal{D}_n} \ell(f(\boldsymbol{x}), y). \tag{2.2}$$

This approach is known as Empirical Risk Minimisation (ERM) [Vap98]. However, ERM is usually avoided in ML as it leads to severe overfitting of the model on the training dataset. As a matter of fact, in this case the training process could choose a model, complicated enough to perfectly describe all the training samples (including noise, which afflicts them). In other words, ERM implies memorisation of data rather than learning from them.

A more effective approach is to minimise a cost function where the trade-off between accuracy on the training data and a measure of the complexity of the selected model is achieved [TA79], adopting regularisation procedures. A common regularisation procedure compels that the solution of an ill-posed optimisation problem satisfies some additional conditions, in order to avoid overfitting the problem. In fact, it is important to learn a solution able to generalise over data, and in order to do so, the sum of the empirical error $\hat{L}^{\mathcal{D}_n}(f)$ and of a regularisation term $C(f)$ is minimised, in order to avoid that particular solution are chosen over more general and explicative ones. As a result, the solution f^* of the problem is found adopting the following equation:

$$f^* : \quad \min_f \quad \hat{L}^{\mathcal{D}_n}(f) + \lambda\, C(f). \tag{2.3}$$

It is important to note that generally $\hat{L}^{\mathcal{D}_n}(f)$ differs from the real $L(f)$ and the difference between them is measured with the generalisation error, which can be seen as the difference between the error on the training set and error on the underlying joint probability distribution.

2.1 Error Measures

The error that f commits, in approximating μ, is measured with reference to a loss function $\ell : \mathcal{X} \times \mathcal{Y} \times \mathcal{F} \to [0, \infty)$ through the empirical error. Generally, for binary classification problems, the hard loss function is used which counts the number of errors $\ell_{\mathrm{H}}(f(\boldsymbol{x}), y) = [f(\boldsymbol{x}) \neq y] \in \{0, 1\}$ [Vap98], while for regression there are typically two losses that are mainly used: the absolute loss $\ell_1(f(\boldsymbol{x}), y) = |f(\boldsymbol{x}) - y|$ and the squared loss $\ell_2(f(\boldsymbol{x}), y) = (f(\boldsymbol{x}) - y)^2$. Based on these losses it is possible to define different indexes of performance, which differently weight the distance between y_i^t and $f(\boldsymbol{x}_i^t)$ [GGA13]:

- the Average Misclassifications Rate (AMR) is the mean number of misclassified samples:

$$\text{AMR} = \frac{1}{m} \sum_{i=1}^{m} \ell_H(f(\boldsymbol{x}_i^t), y_i^t); \tag{2.4}$$

- the Confusion Matrix, which measures four different quantities:
  - TN $= {}^{100}/_{m} \sum_{i=1}^{m} [f(\boldsymbol{x}_i^t = y_i^t \wedge y_i^t = -1]$ which is the percentage of true negative;
  - TP $= {}^{100}/_{m} \sum_{i=1}^{m} [f(\boldsymbol{x}_i^t = y_i^t \wedge y_i^t = +1]$ which is the percentage of true positive;
  - FN $= {}^{100}/_{m} \sum_{i=1}^{m} [f(\boldsymbol{x}_i^t \neq y_i^t \wedge y_i^t = -1]$ which is the percentage of false negative;
  - FP $= {}^{100}/_{m} \sum_{i=1}^{m} [f(\boldsymbol{x}_i^t \neq y_i^t \wedge y_i^t = +1]$ which is the percentage of false positive;
- the Mean Squared Error (MSE) is computed by taking the mean square loss value of f over $\mathcal{T}_m$:

$$\text{MSE} = \frac{1}{m} \sum_{i=1}^{m} \ell_2(f(\boldsymbol{x}_i^t), y_i^t); \tag{2.5}$$

- the Normalized Mean Square Error (NMSE) is similar to the MSE but a normalisation term composed of the mean value of the difference between the true values and their mean is applied:

$$\text{NMSE} = \sum_{i=1}^{m} \frac{\ell_2(f(\boldsymbol{x}_i^t), y_i^t)}{\Delta}; \tag{2.6}$$

- the Relative Error Percentage (REP) is similar to NMSE but the normalisation term is composed of the sum of the squared true values. Then the result is square rooted and reported in percentage:

$$\text{REP} = 100 \sqrt{\frac{\sum_{i=1}^{m} \ell_2(f(\boldsymbol{x}_i^t), y_i^t)}{\sum_{i=1}^{m} (y_i^t)^2}}; \tag{2.7}$$

- the Mean Absolute Error (MAE) is similar to MSE but the absolute loss is used instead:

$$\text{MAE} = \frac{1}{m}\sum_{i=1}^{m}\ell_1(f(\boldsymbol{x}_i^t), y_i^t); \tag{2.8}$$

- the Mean Absolute Percentage Error (MAPE) can be described as the MAE expressed in percentage:

$$\text{MAPE} = \frac{100}{m}\sum_{i=1}^{m}\frac{\ell_1(f(\boldsymbol{x}_i^t), y_i^t)}{y_i^t}; \tag{2.9}$$

- the Pearson Product-Moment Correlation Coefficient (PPMCC) is a measure of the linear dependency between $f(\boldsymbol{x}_i^t)$ and y_i^t with $i \in \{1, \cdots, m\}$:

$$\text{PPMCC} = \frac{\sum_{i=1}^{m}(y_i^t - \overline{y})(f(\boldsymbol{x}_i^t) - \hat{y})}{\sqrt{\sum_{i=1}^{m}(y_i^t - \overline{y})^2}\sqrt{\sum_{i=1}^{n}(f(\boldsymbol{x}_i^t) - \hat{y})^2}}, \tag{2.10}$$

where the following notation is exploited

$$\overline{y} = \frac{1}{m}\sum_{i=1}^{m} y_i^t, \quad \Delta = \frac{1}{m}\sum_{i=1}^{m}(y_i^t - \overline{y})^2, \quad \hat{y} = \frac{1}{m}\sum_{i=1}^{m} f(\boldsymbol{x}_i^t). \tag{2.11}$$

2.2 Model Selection

MS deals with the problem of tuning the hyperparameters of each learning algorithm [AGOR12]. Several methods exist for MS purpose: resampling methods, like the well-known k-Fold Cross Validation (KCV) [K$^+$95] or the non-parametric Bootstrap (BTS) approach [ET94, ABR00] approaches, which represent the state-of-the-art MS approaches when targeting real-world applications. Resampling methods rely on a simple idea: the original dataset $\mathcal{D}_n$ is resampled once or many (n_r) times, with or without replacement, to build two independent datasets called training, and validation sets, respectively $\mathcal{L}_l^r$ and $\mathcal{V}_v^r$, with $r \in \{1, \cdots, n_r\}$. Note that $\mathcal{L}_l^r \cap \mathcal{V}_v^r = \oslash$, $\mathcal{L}_l^r \cup \mathcal{V}_v^r = \mathcal{D}_n$. Then, in order to select the best combination the hyperparameters $\mathcal{H}$ in a set of possible ones $\mathfrak{H} = \{\mathcal{H}_1, \mathcal{H}_2, \cdots\}$ for the algorithm $\mathscr{A}_\mathcal{H}$ or, in other words, to perform the MS phase, the following procedure has to be applied:

$$\mathcal{H}^* : \quad \min_{\mathcal{H}\in\mathfrak{H}} \frac{1}{n_r}\sum_{r=1}^{n_r}\frac{1}{v}\sum_{(\boldsymbol{x}_i, y_i)\in\mathcal{V}_v^r}\ell(\mathscr{A}_{\mathcal{H},\mathcal{L}_l^r}(\boldsymbol{x}_i), y_i), \tag{2.12}$$

where $\mathscr{A}_{\mathcal{H},\mathcal{L}_l^r}$ is a model built with the algorithm $\mathscr{A}$ with its set of hyperparameters $\mathcal{H}$ and with the data $\mathcal{L}_l^r$. Since the data in $\mathcal{L}_l^r$ are independent from the ones in $\mathcal{V}_v^r$, the idea is that $\mathcal{H}^*$ should be the set of hyperparameters which allows to achieve a small error on a data set that is independent from the training set.

If $r = 1$, if l and v are aprioristically set such that $n = l + v$, and if the resample procedure is performed without replacement, the hold out method is obtained [AGOR12]. For implementing the complete k-fold cross validation, instead, it is needed to set $r \leq \binom{n}{k}\binom{n-\frac{n}{k}}{k}$, $l = (k-2)\frac{n}{k}$, $v = \frac{n}{k}$, and $t = \frac{n}{k}$ and the resampling must be done without replacement [K+95, AC10, AGOR12]. Finally, for implementing the BTS, $l = n$ and $\mathcal{L}_l^r$ must be sampled with replacement from $\mathcal{D}_n$, while $\mathcal{V}_v^r$ and $\mathcal{T}_t^r$ are sampled without replacement from the sample of $\mathcal{D}_n$ that have not been sampled in $\mathcal{L}_l^r$ [ET94, AGOR12]. Note that for the BTS procedure $r \leq \binom{2n-1}{n}$.

2.3 Feature Mapping

Generally, the input features of $\mathcal{D}_n$ are subject to a process of transformation. This process is usually referred as Feature Mapping (FM) [SSBD14] since the input features are mapped from an initial input space X to a new input space Ψ. In this way, the FM procedure allows to find a new suitable representation $\boldsymbol{\varphi}(\boldsymbol{x}) \in \Psi$ of the data $\boldsymbol{x} \in \mathcal{X}$ by defining a function $\boldsymbol{\varphi} : \mathcal{X} \to \Psi$ mapping $\boldsymbol{x}$ to a new feature space where it is possible to then learn a simple linear model. FM is usually adopted since the new feature space Ψ could enhance the performance capability of the predictor, depending on the values of $\boldsymbol{x}$ and y.

Several kernel functions can be retrieved in literature [Sch01, CST00], each one with a particular property that can be exploited based on the problem under exam. A simple FM is the polynomial one which results in the following equations:

$$\boldsymbol{\varphi}(\boldsymbol{x}) = \left[\prod_{i=1}^{5d} v_i^{k_i} : \sum_{i=1}^{5d} k_i = j, j \in \{0, 1, \cdots, p\}\right]^T \in \mathbb{R}^{\sum_{j=0}^{p}\binom{5d}{j}} \tag{2.13}$$

$$\boldsymbol{v}=\left[x_1, \cdots, x_d, \frac{1}{x_1}, \cdots, \frac{1}{x_d}, \ln(x_1), \cdots, \ln(x_d), e^{x_1}, \cdots, e^{x_d}, e^{-x_1}, \cdots, e^{-x_d}\right]^T \in \mathbb{R}^{5d}.$$

The problem of the FM of Eq.(2.13) is that its computation is unfeasible if d or p are large since the problem is NP-hard. Nevertheless, in these cases, it is possible to exploit the kernel trick and noting that the proposed FM can be expressed with a simple polynomial kernel

$$\boldsymbol{\varphi}(\boldsymbol{a})^T\boldsymbol{\varphi}(\boldsymbol{b}) = (\boldsymbol{v_a}^T\boldsymbol{v_b} + c)^p = K(\boldsymbol{a}, \boldsymbol{b}) \tag{2.14}$$

$$\boldsymbol{v_a}=\left[a_1, \cdots, a_d, \frac{1}{a_1}, \cdots, \frac{1}{a_d}, \ln(a_1), \cdots, \ln(a_d), e^{a_1}, \cdots, e^{a_d}, e^{-a_1}, \cdots, e^{-a_d}\right]$$

$$\boldsymbol{v_b}=\left[b_1, \cdots, b_d, \frac{1}{b_1}, \cdots, \frac{1}{b_d}, \ln(b_1), \cdots, \ln(b_d), e^{b_1}, \cdots, e^{b_d}, e^{-b_1}, \cdots, e^{-b_d}\right],$$

it is possible to avoid the NP-hard problem and use the desired FM. Note that $p \in \{0, 1, \cdots\}$ is the desired degree of the polynomial and $c \in [0, \infty)$ is a parameter trading off the influence of

higher-order versus lower-order terms in the polynomial. p and c together with λ are hyperparameters that need to be tuned in order to optimise the performance of the final model.

Besides the polynomial FM, it is possible to define other possible FMs such as the Gaussian one:

$$\boldsymbol{\varphi}(\boldsymbol{x}) = \exp\left(-\frac{\boldsymbol{x}^2}{2\sigma^2}\right)\left[1, \sqrt{\frac{1}{1!\sigma^2}}\boldsymbol{x}, \sqrt{\frac{1}{2!\sigma^4}}\boldsymbol{x}^2, \cdots\right]^T, \tag{2.15}$$

which can be mapped to a Kernel function similarly to the polynomial FM seen before:

$$\boldsymbol{\varphi}(\boldsymbol{a})^T\boldsymbol{\varphi}(\boldsymbol{b}) = (\boldsymbol{v_a}^T\boldsymbol{v_b}) = K(\boldsymbol{a}, \boldsymbol{b}) = \exp\left(\frac{-||\boldsymbol{a}-\boldsymbol{b}||_2^2}{2\sigma^2}\right), \tag{2.16}$$

where $2\sigma^2$ is the hyperparameter of the kernel, measuring how much the Gaussian function is smooth. Sometimes, $1/2\sigma^2$ is defined as γ.

It is important to note that, the Gaussian kernel, differently from the polynomial one, is the product of two $\boldsymbol{\varphi}$ which are not finite, so the dimension of the final space into which data points are mapped is infinite. This characteristic make the Gaussian Kernel a suitable FM mechanism in many applications.

2.4 Supervised Learning Algorithms

Many algorithms $\mathscr{A}_{\mathcal{H}}$ can be used in regression and classification techniques in order to obtain f. In particular, Neural Networks (NNs), Kernel Methods (KMs), Ensemble Methods (EMs) are among the most used and performing.

NNs are ML techniques which combine many simple models of a human brain neuron, called perceptrons [Ros58], in order to build a complex network [Bis95, RHW88]. The neurons are organised in stacked layers connected together by weights that are learned based on the available data via back-propagation [RHW88]. An NN is characterised by the number of layers and the number of neurons for each layer of which it is composed. If the architecture of the NN consists of only one hidden layer, it is called shallow (SNN), while, if multiple layers are staked together, the architecture is defined as deep (DNN) [HOT06, Ben09, BCV13]. Extreme Learning Machines (ELM) are a particular kind of SNN, where the weights of the first layer are randomly chosen while the ones of the output layers are computed according to the Regularised Least Squares principle.

KMs are a family of ML techniques which exploits the "Kernel trick" for distances in order to extend linear techniques to the solution of non-linear problems [STC04]. KMs approximate μ as the function which minimises the trade-off between the sum of the accuracy over the data, namely

the empirical error, and the complexity of the solution, namely the regularisation term. Some of the most known and effective KM techniques are Kernelized Regularised Least Squares (KRLS), Support Vector Machines (SVM), and Support Vector Regression (SVR) [RYP03, KL03]. These methods are characterised by the kernel adopted (e.g: linear, Gaussian) and by the constant balancing the trade-off between the regularisation term and the error over data.

EMs techniques relies on the simple fact that combining the output of several classifiers results in a much better performance than using any one of them alone [Bre01]. Random Forest and Random Rotation Ensembles, the-art and widely adopted methods, combine many decision trees in order to obtain an effective predictor which has limited hyperparameter sensitivity and high numerical robustness.

Among the other state-of-the-art ML, it is possible to recall also Bayesian Methods (BMs), and Lazy Methods (LMs).

BMs are ML techniques where, instead of choosing a particular $f \in \mathcal{F}$ a distribution for choosing $f \in \mathcal{F}$ is defined [GCSR14]. Gaussian Processes (GP) learning algorithm is a popular BM [Ras06] which employs a collection of Gaussians in order to compute the posterior distribution of the $f(\boldsymbol{x})$. In fact, this algorithm defines the probability distribution of the output values as a sum of Gaussians whose variance is fixed according to the training data. The hyperparameter of the GP $\mathcal{H}^{\text{GP}}$ is the parameter which governs the Gaussians width h_1.

LMs ML techniques are learning method in which the definition of f is delayed until $f(\boldsymbol{x})$ needs to be computed [Duc00]. LMs approximate μ locally with respect to $\boldsymbol{x}$. K-Nearest Neighbours (KNN) is one of the most popular LM due to its implementation simplicity and effectiveness [CH67]. The hyperparameter of the KNN $\mathcal{H}^{\text{KNN}}$ is the number of neighbours of $\boldsymbol{x}$ to be considered h_1.

2.4.1 Neural Networks

Every NN differentiates itself by the architecture of layers of neurons of which they are composed, dependently to the specific task they have to accomplish.

The simplest NN type is the one represented by single-hidden-layer feed-forward NNs usually referred as Shallow NN (SNN):

$$f(\boldsymbol{x}) = \sum_{i=1}^{h} w_i g_i(\boldsymbol{x}), \tag{2.17}$$

where $g_i : \mathbb{R}^d \to \mathbb{R},\ i \in \{1, \cdots, h\}$ is the hidden-layer output corresponding to the input sample $\boldsymbol{x} \in \mathbb{R}^d$, and $\boldsymbol{w} \in \mathbb{R}^h$ is the output weight vector between the hidden layer and the output layer. In this case, the input layer has d neurons and connects to the hidden layer (having h neurons) through a set of weights $W \in \mathbb{R}^{h \times (0, \cdots, d)}$ and a nonlinear activation function, $\varphi : \mathbb{R} \to \mathbb{R}$. Thus,

the i-th hidden neuron response to an input stimulus $\boldsymbol{x}$ is:

$$g_i(\boldsymbol{x}) = \varphi\left(W_{i,0} + \sum_{j=1}^{d} W_{i,j}x_j\right). \tag{2.18}$$

By adding layers to the SNN model, it is possible to obtain a DNN, where the response $f(\boldsymbol{x})^k$ of layer k uses as input the output of $k-1$ layer:

$$f(\boldsymbol{x})^k = \sum_{i=1}^{h_k} w_i^k g_i(f(\boldsymbol{x})^{k-1}). \tag{2.19}$$

DNNs, as one can easily understand from the previous description, are characterised by many hyperparameters, which deeply influence the representation and generalisation abilities of the architecture. In particular, the hyperparameters of a DNN are [HS06]:

- the activation function (e.g. sigmoidal, hyperbolic tangent, and rectified linear);
- the number of layers;
- the number of neurons for each layer;
- the type of regularises and magnitude of regularisation (e.g. norm of the weights, dropout, and early stopping);
- the loss function (e.g. quadratic and linear);
- the optimiser and optimisation time (e.g. stochastic gradient descent and mini-batch gradient descent).

Moreover, after fixing the above mentioned hyperparameters the resulting architecture depends also on the initialisation of the weights of the network [GGHGGM$^+$05]. Note that, in a DNN, this last variability is not so pronounced if the hyperparameters have been appropriately set.

Consequently, in order to tune these hyperparameters, it is necessary to adopt a reliable MS strategy [AGOR12]. A common approach for tuning the hyperparameters of a leaning algorithm is to build a grid of possible configuration of hyperparameters, or to randomly select a subset of possible configurations [BB12], and then select the best one according to the KCV [AGOR12] of the BTS [ET94]. Unfortunately, in a DNN, the number of hyperparameters is too high to perform this task effectively and, for this reason, the configurations are often chosen based on the experience of the data scientist and with a bit of data snooping [Whi00]. This approach, even if commonly exploited, may lead to significant biases in estimation [HU14]. For this reason it is always necessary to keep apart a set of unused data, the test set, that can be seen just once, in order to test the validity of the applied procedure and selected architecture in order to report unbiased results [Vap98, Whi00, HU14].

Recently, many advances have been made in this field or research by developing new neurons [LBBH98], new activation functions [HSM$^+$00], new optimisation techniques [NCL$^+$11],

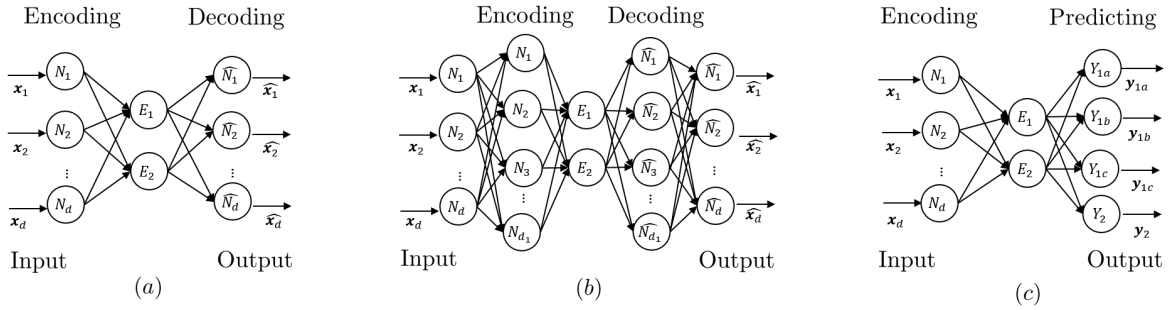


Figure 2.1: Autoencoder.

new regularisation methods in order to reduce the overfitting tendency of complex and deep networks [SHK+14]. These advances allowed the researcher to successfully apply these methods on increasingly different and difficult real world problems. In particular, DNN have been shown to be an extremely powerful tool for Feature Selection (FS) and extraction purposes, both in the supervised and unsupervised context [Ben09, BCV13, HOT06, HS06]. In fact, before learning, feature learning is required and conducted in many applications to achieve a satisfactory accuracy [GE03, Sch15]. DNN represents a state-of-the-art choice in this context [BCV13, HOT06]. The deep architecture extracts features through a multilayered feature representation framework where the higher layers represent more abstract information than those from the lower ones.

2.4.1.1 Autoencoders

By taking advantage of the natural input-filtering ability of NNs, DNNs can be used in order to perform an unsupervised learning FS process. In fact, DNNs, instead of learning the relationship between the input space $\mathcal{X}$ and the output space $\mathcal{Y}$, can be used to perform an often lower [BK88] (but sometimes higher [HOT06]) dimensional FM, which is able to explain, in a more informative way, the point sampled from $\mathcal{X}$. This architecture is called autoencoder [HZ94] and is depicted in Figure 2.1.A. By staking many autoencoders it is possible to obtain a deep autoencoder [HS06], depicted in Figure 2.1.B. The stacked autoencoders can be learned incrementally by adding and learning one simple autoencoder at a time in order to avoid the gradient vanishing effect [Sch15]. Once the representation has been learned, it is possible to use the latter in order to learn the relation between $\mathcal{X}$ and $\mathcal{Y}$ with the final DNN architecture depicted in Figure 2.1.C [HOT06]. The final architecture uses the unsupervised learned representation as starting point, which is then fine-tuned based on the desired target tasks.

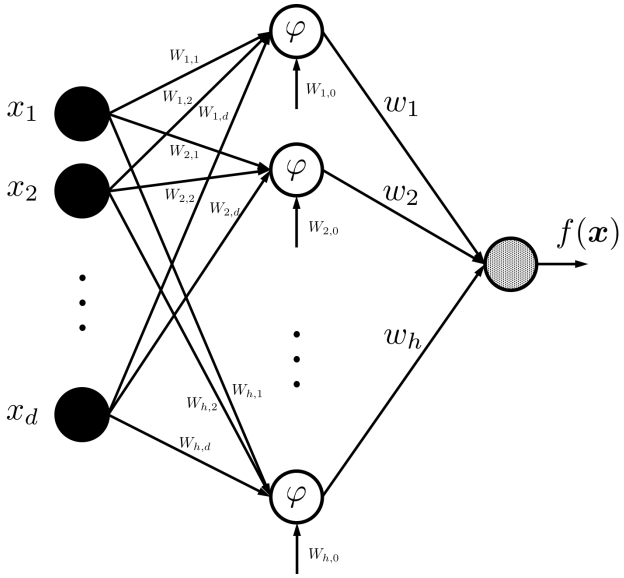


Figure 2.2: SELM structure.

2.4.1.2 Extreme Learning Machines

Another notable NN, is the one defined as ELM, which is present in literature both in it shallow version (SELM) and in its deep version (DELM). ELM [CH13, HHSY15, HZS06] were introduced to overcome problems posed by back-propagation training algorithm [Hua14, Hua15, RRZ97, RHW88]: potentially slow convergence rates, critical tuning of optimisation parameters, and presence of local minima that call for multi-start and re-training strategies. The original ELM are also called SELM because they have been developed for the single-hidden-layer feedforward NNs [HLCS08, HCS06, HZS04], and they have been generalised in order to cope with cases where ELM are not neuron alike. SELM were later improved to cope with problems intractable by shallow architectures [Ben09, BCV13, VLBM08, HOT06, ZHL$^+$15], by proposing various DELM built upon a deep architecture [KZHV13, TDH16, TM16, OFC$^+$17], in order to make possible to extract features by a multilayer feature representation framework.

In SELM, the parameters of the first layer W are set randomly and a vector of weighted links, $\boldsymbol{w} \in \mathbb{R}^h$, connects the hidden neurons to the output neuron without any bias. The overall output function of the network (see Figure 2.2) is:

$$f(\boldsymbol{x}) = \sum_{i=1}^{h} w_i \varphi \left(W_{i,0} + \sum_{j=1}^{d} W_{i,j} x_j \right) = \sum_{i=1}^{h} w_i \varphi_i(\boldsymbol{x}). \tag{2.20}$$

It is convenient to define an activation matrix, $A \in \mathbb{R}^{n \times h}$, such that the entry $A_{i,j}$ is the activation

value of the j-th hidden neuron for the i-th input pattern. The A matrix is:

$$A = \begin{bmatrix} \varphi_1(\boldsymbol{x}_1) & \cdots & \varphi_h(\boldsymbol{x}_1) \\ \vdots & \ddots & \vdots \\ \varphi_1(\boldsymbol{x}_n) & \cdots & \varphi_h(\boldsymbol{x}_n) \end{bmatrix}. \tag{2.21}$$

In the SELM model the weights W are set randomly and are not subject to any adjustment, and the quantity $\boldsymbol{w}$ in Eq. (2.20) is the only degree of freedom. Hence, the training problem reduces to a simple Regularised Least Squares (RLS) problem [CDV07]:

$$\boldsymbol{w}^* = \arg\min_{\boldsymbol{w}} \|A\boldsymbol{w} - \boldsymbol{y}\|^2 + \lambda \|\boldsymbol{w}\|^2, \tag{2.22}$$

where $\lambda \in [0, \infty)$ is a hyperparameter that must be tuned during the Model Selection (MS) phase [One18], since it balances the trade off between accuracy complexity of the model measures with the square loss and the $L2$ regularizer respectively. Consequently, the vector of weights $\boldsymbol{w}^*$ is then obtained as follows:

$$\boldsymbol{w}^* = (A^T A + \lambda I)^+ A^T \boldsymbol{y}, \tag{2.23}$$

where $I \in \mathbb{R}^{h \times h}$ is an identity matrix and $(\cdot)^+$ is the Moore-Penrose matrix pseudoinverse. Note that h, the number of hidden neurons, is another hyperparameter that needs to be tuned based on the problem under exam. Note also that other regularisers can be exploited (e.g. sparse regularisers [Tib96, ZH05]).

Due to its shallow architecture, feature learning using SELM may not be effective even when h is large. Since feature learning is often useful to improve the accuracy of the final model, multilayer (deep) solutions are usually needed. In [KZHV13, TDH16] multilayer learning architectures are developed using ELM-based autoencoder (AE) as its building block (see Figure 2.3), which results in a sort of DELM. At each layer i of the l layers, each one composed of $h_{i \in \{1,\cdots,l\}}$ neurons, the DELM tries to reconstruct the input data and the outputs of the previous layer are used as the inputs of the next one. Basically, instead of having just one output, a series of outputs $\hat{x}_j$ with $j \in \{1, \cdots, d\}$ is obtained such that

$$\hat{x}_j = f_j(\boldsymbol{x}) = \sum_{i=1}^{h} w_{i,j} \varphi \left(W_{i,0} + \sum_{j=1}^{d} W_{i,j} x_j \right) = \sum_{i=1}^{h} w_{i,j} \varphi_i(\boldsymbol{x}), \tag{2.24}$$

where $w_{i,j}$ with $i \in \{1, \cdots, h\}$ are found with the same approach of SELM. Before the supervised regularised least mean square optimisation, the encoded outputs are directly fed into the last layer for decision-making, without random FM. Differently from SELM, DELM do not require fine-tuning for the entire system and consequently the training speed can be much faster than the traditional back propagation based Deep Learning. Training the DELM is equivalent to training many SELM. Consequently, it is possible to take advantage of a deep architecture by exploiting only the optimisation tools presented for the SELM.

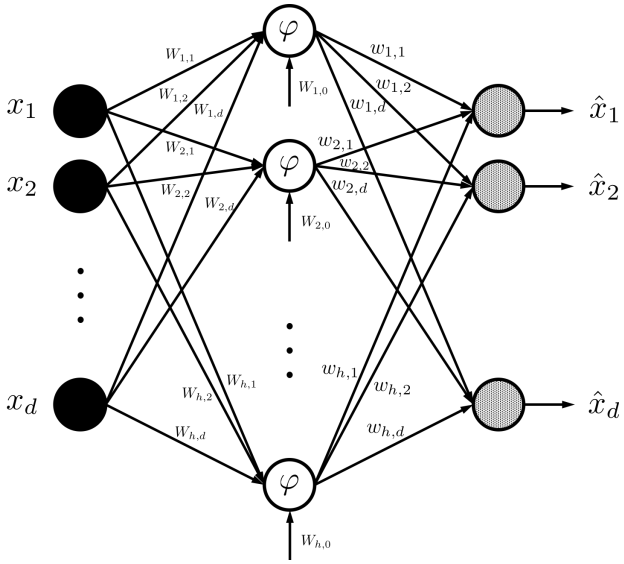


Figure 2.3: DELM AE block.

2.4.2 Kernel Methods

According to the common KMs definition, all KMs are characterised by a kernel function which enables them to transfer the original input data in a high-dimensional, implicit feature space without computing the exact coordinates of that data in that space. In details, KMs try to minimize a cost function (see Eq. 2.3) where the trade-off between accuracy on the training data and a measure of the complexity of the selected model is achieved [TA79], implementing the simple Occam's razor principle of Eq. 2.3.

In other words, the best approximating function f^* is chosen as the one that is complicated enough to learn from data without overfitting them. Referring to Eq. 2.3, $C(\cdot)$ is a complexity measure: depending on the exploited ML approach, different measures are realised. Instead, $\lambda \in [0, \infty)$ is a hyperparameter, that must be aprioristically set and is not obtained as an output of the optimisation procedure: it regulates the trade-off between the overfitting tendency, related to the minimisation of the empirical error, and the underfitting tendency, related to the minimisation of $C(\cdot)$. The optimal value for λ is problem-dependent, and tuning this hyperparameter is a non-trivial task, as it is discussed later in this section. In KMs, the models are defined as

$$f(\boldsymbol{x}) = \boldsymbol{w}^T \boldsymbol{\varphi}(\boldsymbol{x}), \tag{2.25}$$

where $\boldsymbol{\varphi}$ is an aprioristically defined FM [SSBD14], which strongly depends on the particular problem under exam, allowing to keep the structure of $f(\boldsymbol{x})$ linear.

It is important to note that the different KMs distinguish themselves by the shape of their loss function, or, to make it simple, by how to weight and consider the committed errors. In Figure 2.4

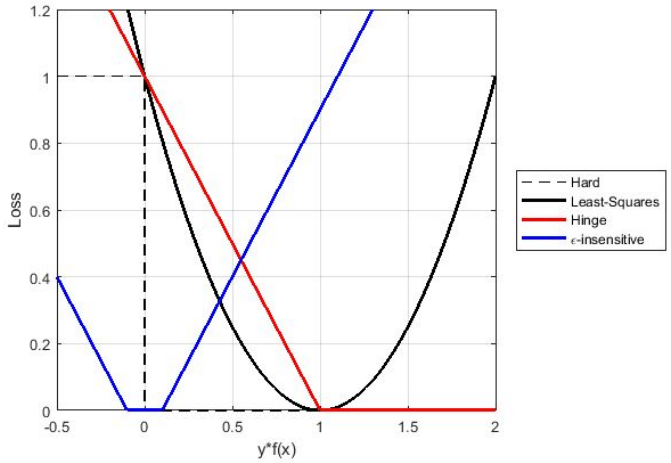


Figure 2.4: Different Loss functions shapes.

the three different loss function are reported for KRLS (least-squares loss), SVM (hinge loss), and SVR (ϵ-insensitive loss).

2.4.2.1 Kernel Regularised Least Squares

Among the different KMs, KRLS can be considered as one of the simplest to implement. In fact, the complexity of the models, in KRLS, is measured as

$$C(f) = \|\boldsymbol{w}\|^2, \tag{2.26}$$

i.e. the Euclidean norm of the set of weights describing the regressor, which is a quite standard complexity measure in ML [SSBD14]. By adopting the square loss, Problem (2.3) can be reformulated as

$$\boldsymbol{w}^* : \quad \min_{\boldsymbol{w}} \quad \sum_{i=1}^{n} \left[\boldsymbol{w}^T \boldsymbol{\varphi}(\boldsymbol{x}_i) - y_i\right]^2 + \lambda \|\boldsymbol{w}\|^2. \tag{2.27}$$

By exploiting the Representer Theorem [SHS01], the solution f^* of the RLS Problem (2.27) can be expressed as a linear combination of the samples projected in the space defined by $\boldsymbol{\varphi}$

$$f^*(\boldsymbol{x}) = \sum_{i=1}^{n} \alpha_i \boldsymbol{\varphi}(\boldsymbol{x}_i)^T \boldsymbol{\varphi}(\boldsymbol{x}). \tag{2.28}$$

It is worth remembering from 2.3 that, according to the kernel trick, it is possible to reformulate $f^*(\boldsymbol{x})$ without an explicit knowledge of $\boldsymbol{\varphi}$, and consequently avoiding the course of dimension-

ality of computing $\boldsymbol{\varphi}$, by using a proper kernel function $K(\boldsymbol{x}_i, \boldsymbol{x}) = \boldsymbol{\varphi}(\boldsymbol{x}_i)^T \boldsymbol{\varphi}(\boldsymbol{x})$

$$f^*(\boldsymbol{x}) = \sum_{i=1}^{n} \alpha_i K(\boldsymbol{x}_i, \boldsymbol{x}). \tag{2.29}$$

The KRLS problem of Eq. (2.27) can be reformulated by exploiting kernels as

$$\boldsymbol{\alpha}^* : \quad \min_{\boldsymbol{\alpha}} \quad \|Q\boldsymbol{\alpha} - \boldsymbol{y}\|^2 + \lambda \boldsymbol{\alpha}^T Q \boldsymbol{\alpha}, \tag{2.30}$$

where $\boldsymbol{y} = [y_1, \ldots, y_n]^T$, $\boldsymbol{\alpha} = [\alpha_1, \ldots, \alpha_n]^T$, the matrix K such that $Q_{i,j} = K_{ji} = K(\boldsymbol{x}_j, \boldsymbol{x}_i)$, and $I \in \mathbb{R}^{n \times n}$ is the identity matrix. By setting equal to zero the gradient with respect to $\boldsymbol{\alpha}$ it is possible to state that

$$(K + \lambda I)\, \boldsymbol{\alpha}^* = \boldsymbol{y}, \tag{2.31}$$

that is a linear system for which effective solvers have been developed throughout the years, allowing coping with even very large sets of training data [You03].

It is important to note that KRLS can be both used for regression and classification tasks, keeping in mind that to obtain a classification label the sign of $f(\boldsymbol{x})$ need to be taken.

2.4.2.2 Support Vector Machines

Differently from KRLS, Support Vector Machines (SVM) can be considered a valuable KMs only for classification purposes. In the simplest binary classification formulation, SVM tries to find the best hyperplane defined by $\boldsymbol{w}$ weights separating two classes of points. This hyperplane is subject to a constraint since it should be the one maximising the distance between the points of the two different classes which are closest to the opposite class, usually referred as support points. The two hyperplanes passing from these support points are referred as support vectors and they are equidistant from the separator hyperplane.

In order to avoid overfitting and add relaxation to the bond, a regularization term is adopted adopting slack variables ξ, which allow to commit some error terms in order to have a smooth separator. As a result the resulting cost function is:

$$\boldsymbol{w}^*, \boldsymbol{\xi}^* = \quad \min_{\boldsymbol{w}} \frac{||\boldsymbol{w}||^2}{2} + \lambda \sum_{i=1}^{n} \xi_i \tag{2.32}$$

$$y_i - \boldsymbol{w}\boldsymbol{\varphi}(\boldsymbol{x}_i) \leq 1 + \xi_i, \tag{2.33}$$

$$\xi_i \geq 0 \quad i = 1, \cdots, n. \tag{2.34}$$

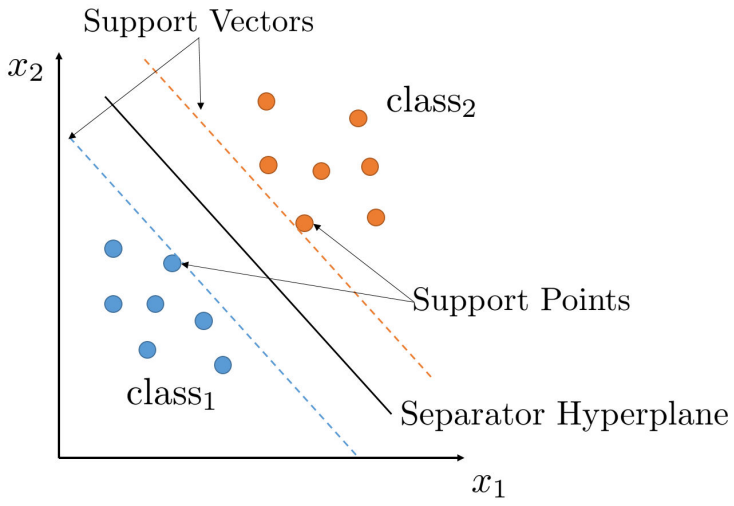


Figure 2.5: 2D representation of SVM separator hyperplane, support vectors, and support points.

It is possible to re-write the problem with its Primal Lagrangian form

$$\mathscr{L}_p(\boldsymbol{w}, \boldsymbol{\xi}, \boldsymbol{\alpha}, \boldsymbol{\mu}) = \frac{||\boldsymbol{w}||^2}{2} + \lambda \sum_{i=1}^{n} \xi_i - \sum_{i=1}^{n} \alpha_i (y_i - \boldsymbol{w}\boldsymbol{\varphi}(\boldsymbol{x}_i) - 1 + \xi_i) - \sum_{i=1}^{n} \mu_i \xi_i, \tag{2.35}$$

$$\alpha_i \geq 0, \mu_i \geq 0 \quad i = 1, \cdots n, \tag{2.36}$$

in order to find a solution also satisfying the bonds. So, it is possible to define a Dual Lagrangian [Vap98] by computing the solutions of the Karush-Kuhn-Tucker (KKT) conditions and use them and the bonds to define $\boldsymbol{w}$, b, and $\boldsymbol{\xi}$ in terms of $\boldsymbol{\alpha}$ and $\boldsymbol{\mu}$:

$$\frac{\delta \mathscr{L}}{\delta \boldsymbol{w}} = \boldsymbol{w} - \sum_{i=1}^{n} \alpha_i y_i \boldsymbol{\varphi}(\boldsymbol{x}) = 0, \tag{2.37}$$

$$\frac{\delta \mathscr{L}}{\delta \xi_i} = \lambda - \alpha_i - \mu_i = 0. \tag{2.38}$$

The resulting Dual Lagrangian expression is:

$$\mathscr{L}_d(\boldsymbol{\alpha}) = \frac{1}{2} \sum_{i,j=1}^{n} \alpha_i \alpha_j y_i y_j \boldsymbol{\varphi}(\boldsymbol{x}_i)^T \boldsymbol{\varphi}(\boldsymbol{x}_j) + \sum_{i=1}^{n} \alpha_i, \tag{2.39}$$

which lead to the following convex problem

$$\min_{\boldsymbol{\alpha}} \mathscr{L}_d(\boldsymbol{\alpha}) = \frac{1}{2} \sum_{i,j=1}^{n} \alpha_i \alpha_j y_i y_j \boldsymbol{\varphi}(\boldsymbol{x}_i)^T \boldsymbol{\varphi}(\boldsymbol{x}_j) + \sum_{i=1}^{n} \alpha_i, \tag{2.40}$$

$$0 \leq \alpha_i \leq \lambda. \tag{2.41}$$

It is important to note that $\boldsymbol{\varphi}(\boldsymbol{x}_i)^T\boldsymbol{\varphi}(\boldsymbol{x}_j)$ can be rewritten adopting a Kernel $K(\boldsymbol{x}_i, \boldsymbol{x}_j) \leq \boldsymbol{\varphi}(\boldsymbol{x}_i), \boldsymbol{\varphi}(\boldsymbol{x}_j) >$ function as for KRLS and that the minimumm of dual Lagrangian can be easily found adopting a quadratic programming procedure. Similarly from KRLS, the result of this minimization process allows to find the best $\boldsymbol{\alpha}$ which, for the Representer's Theorem, define the function of the classifier.

$$f(\boldsymbol{x}) = \sum_{i=1}^{n} \alpha_i y_i K(\boldsymbol{x}_i, \boldsymbol{x}). \tag{2.42}$$

2.4.2.3 Support Vector Regression

Since SVM was defined in order to deal with classification problems, a slight variation of its concept need to be adopted in order to target regression tasks. To this purpose Support Vector Regression (SVR) was defined. SVR's method base formulation tries to calculate the hyper plane minimizing a Tychonov regularization problem composed by a term indicating the complexity of the hyper planes parameters and a term summing the errors committed by the model. This method defines an ϵ-tube around the hyper plane in which the samples should be included as much as possible; if a point lies inside this ϵ-tube, it is not considered as an error point. In facts, the regularisation term is used in order to allow the possibility of committing an error, which is evaluated using the ϵ-insensitivity loss function. The equations of the problem thus formulated are:

$$\min_{\boldsymbol{w}} \frac{||\boldsymbol{w}||^2}{2} + \lambda \sum_{i=1}^{n} (\xi_i - \hat{\xi}_i) \tag{2.43}$$

$$y_i - \boldsymbol{w}\boldsymbol{\varphi}(\boldsymbol{x}_i) \leq \epsilon + \xi_i, \tag{2.44}$$

$$\boldsymbol{w}\boldsymbol{\varphi}(\boldsymbol{x}_i) - y_i \leq \epsilon + \hat{\xi}_i, \tag{2.45}$$

$$\xi, \hat{\xi}_i \geq 0 \quad i = 1, \cdots, n. \tag{2.46}$$

$$\tag{2.47}$$

The solution to this minimisation problem is achieved by calculating the primal and dual Lagrangian, and since the equations and the bounds are all convex, the solution to be found, the values of $\boldsymbol{\alpha}$ and $\hat{\boldsymbol{\alpha}}$ which minimise the cost function, is unique. The definition of the KKTs give us the value of $\boldsymbol{w} = \sum_{i=1}^{n} (\alpha_i - \hat{\alpha}_i)\boldsymbol{x}_i$, which leads to the definition of the regression function here reported:

$$f(\boldsymbol{x}) = \sum_{i=1}^{n} (\alpha_i - \hat{\alpha}_i)\boldsymbol{\varphi}(\boldsymbol{x}_i)^T\boldsymbol{\varphi}(\boldsymbol{x}). \tag{2.48}$$

This kind of solution can be extended in order to treat non-linear regression problems, adopting the Kernel trick as for SVM and KRLS. The equation of the solution thus becomes:

$$f(\boldsymbol{x}) = \sum_{i=1}^{n} (\alpha_i - \hat{\alpha}_i)K(\boldsymbol{x}_i, \boldsymbol{x}). \tag{2.49}$$

2.4.3 Ensemble Methods

EMs are considered state-of-the-art ML methods since they are characterised by a high numerical robustness, the native capacity of dealing with numerical and categorical features, and effectiveness in many real-world regression and classification problems [Bre01, BF15]. Among the EMs, two main methods distinguish themselves, Random Forests (RFs) and Random Rotation Ensembles (RREs), which can be considered a modification of the original RF formulation.

In the original RF learning phase, the RF is composed by n_t trees $\{T_1, \cdots, T_{n_t}\}$ which are built each one independently from all the others [Bre01]. From $\mathcal{D}_n$, $\lfloor bn \rfloor$ samples are sampled with replacement and $\mathcal{D}'_{\lfloor bn \rfloor}$ is built. A tree is constructed of $\mathcal{D}'_{\lfloor bn \rfloor}$ but the best split is chosen among a subset of n_v features over the possible d features randomly chosen at each node. The tree is grown until the node contains a maximum of n_l samples. During the classification phase of a previously unseen X, each tree classifies X in a class $Y_{i \in \{1,\cdots,n_t\}}$, and then the final classification is the $\{p_1, \cdots, p_{n_t}\}$-weighted combination of all the answers of each tree of the RF. The empirical error of the tree T built based on $\mathcal{D}'_{\lfloor bn \rfloor}$ over the out of bag data $\mathcal{D}_n \setminus \mathcal{D}'_{\lfloor bn \rfloor}$ is defined as $\widehat{L}^{\text{oob}}(T)$. $p_{\{i \in 1,\cdots,n_t\}}$ are of paramount importance for the accuracy of an ensemble classifier [LLST13, GLLF15], and for this reason, a state-of-the-art alternative can be exploited as proposed in [Cat07] and recently further developed in [LLST13, OOA16] where $p_i = e^{-\gamma \widehat{L}^{\text{oob}}(T)}$ with $\gamma \in [\gamma, \infty)$. If $\gamma = 0$, $b = 1$, $n_v = \sqrt{n}$, and $n_l = 1$ the original RF formulation is obtained [Bre01].

RF have been recently improved in RREs [BF15] which propose to avoid the initial bootstrapping and the subset FS at each node construction of the trees by replacing it with a random rotation of the numerical feature space before learning each tree of the forest. Note that, since rotations can be sensitive to scale in general and to outliers in particular, the RF developed in [BF15] need to scale the numerical feature space. As suggested by the results in [BF15], the simple scaling of each feature in the range $[0, 1]$ should be adopted. The RF learning algorithm reported in Algorithm 1 is proposed, which merges the original RF formulation [Bre01] with the most recent one of [BF15]. In particular, the learning strategy proposed in [Bre01] can be perfectioned also including the random rotation proposed in [BF15] in order to get the benefits of both approaches. Note that in Algorithm 1 the rotation does not change at each tree but every n_r trees to reduce the computational requirements of the RF with respect to [BF15]. EMs can also be used in order to perform an analysis of each $\mathcal{X}$ feature relevance. This capability can be very useful in all cases where either the original dimensionality of $\mathcal{X}$, or the final dimensionality reached by the FM, is very high. In fact, in the latter case, many unnecessary features could have been generated, which are either redundant or not informative.

In this case a FS phase is required in order to increase the generalisation performance of the model by selecting only the most informative features which best represent the problem to be solved and discarding the others [GE03]. This process allows detecting if the importance of those features, that are known to be relevant for the current problem. The failure of the statistical model

Algorithm 1: RF learning algorithm: learning and forward phases.

/* Learning phase */
Input: $\mathcal{D}_n$, n_t, γ, b, n_v, n_r, and n_l
Output: A set of tree $\{T_1, \cdots, T_{n_t}\}$
for $i \leftarrow 1$ **to** n_t **do**
  if $i - \lfloor i/n_r \rfloor n_r = 1$ **then**
    $\Theta =$ random rotation matrix defined in [BF15];
    $\mathcal{D}_n^r =$ rotate the numerical features space $\mathcal{D}_n$ based on Θ;
  $\mathcal{D}'_{\lfloor bn \rfloor}$ sample with replacement $\lfloor bn \rfloor$ sample from $\mathcal{D}_n^r$;
  $T_i.\Theta = \Theta$; $T_i.T = \mathbf{DT}(\mathcal{D}'_{\lfloor bn \rfloor}, n_v, n_l)$; $T_i.p = Exp[-\gamma \widehat{L}^{\text{oob}}(T_i.T)]$;
/* Forward phase */
Input: X, n_t
Output: Y
for $i \leftarrow 1$ **to** n_t **do**
  $X^r =$ rotate X based on $T_i.\Theta$; $Y_i = T_i.T(X^r)$;
if *Classification Task* **then** $Y = \arg\max_{j \in \{1,\cdots,c\}} \sum_{i \in \{1,\cdots,n_t\}: Y_i = j} T_i.p$;
if *Regression Task* **then** $Y = \sum_{i=1}^{n_t} Y_i \cdot T_i.p$;
/* Functions */
function $T = \mathbf{DT}(\mathcal{D}_n, n_v, n_l)$;
if $n \leq n_l$ **then**
  $T.l =$ mode$(\{Y \in \mathcal{D}_n\})$;
else
  Split $\mathcal{D}_n$ in $\mathcal{D}'_{n'}$ and $\mathcal{D}''_{n''}$ based on the best split over a random subset of size n_v of all the features ;
  $T.s = s$; $T.T' = \mathbf{DT}(\mathcal{D}'_{n'}, n_v, n_l)$; $T.T'' = \mathbf{DT}(\mathcal{D}''_{n''}, n_v, n_l)$;

to properly account for the relevant features might indicate poor quality in the measurements or spurious correlations. FS therefore represents an important step of model verification, since it should generate consistent results with the available knowledge of the system under exam.

Among the different algorithms to perform FS, the EMs can be used to perform a very stable FS procedure. The procedure is a combination of EMs, together with the permutation test [Goo13], in order to perform the selection and the ranking of the features. In details, for every tree, two quantities are computed: the first one is the error on the out-of-bag samples as they are used during prediction, while the second one is the error on the out-of-bag samples after a random permutation of the values of variable j. These two values are then subtracted and the average of the result over all the trees in the ensemble is the raw importance score for variable j (mean decrease in accuracy). This procedure was adopted since it can be easily carried out during the main prediction process inexpensively.

2.5 Anomaly Detection Algorithms

Even if Supervised Learning Techniques have proved to be valuable instruments for the data scientist in many practical applications, they bring some side problems. In practice, retrieving the state or the label of the different target features is a complex operation. It may require the intervention of an experienced operator to manually label them or, in some cases, to stop the activities in order to collect some samples. In both these situations, the cost of labelling the data in order to provide them to a supervised model is too elevated and brings inefficiencies in the process.

To solve these issue, it is possible to look at the same problem from another perspective. For some cases, it is reasonable to state that, for the vast majority of the time, a desirable value for the target features is known, while other undesired states can be defined. If the target features stays in its ordinary operating value the most of the time, it should keep a different value only in few occasions. This new problem can be straightforwardly mapped into a classical outlier (novelty) detection problem [Haw80, STC04, SMS$^+$16] where the aim is to detect unexpected behaviour in the data. This method does not require to know either the actual state of decay of the components, as a regression task would do, or the less detailed information about "desired" or "undesired", as in the binary classification framework. These kinds of DDMs try to build a model of the "usual" operational profile and automatically detect if the data collected "deviating too much" from the established behaviour.

Similarly to their supervised counterpart, also unsupervised ML methods can be divided into different families. Some of the most known and effective techniques for solving these problems according to [SMS$^+$16]. In particular [SMS$^+$16] shows that two anomaly detection methods based on SVM and KNN respectively, are the top choices in this context.

In particular the Global KNN (GKNN), inspired by the KNN, has been originally introduced as an unsupervised distance-based outlier detection method [RRS00, SMS$^+$16]. The hyperparameter GKNN $\mathcal{H}^{\text{GKNN}}$ is the same as the one of KNN.

Another semi-supervised method is One-Class SVM (OCSVM), which is a boundary-based anomaly detection method, inspired by SVM, which enclose the inlier class in a minimum volume hypersphere by minimizing a Tikhonov regularization problem, similar to the one reported for SVM framework. The classical SVM formulation then changes by considering the following minimisation equation:

$$R^*, \boldsymbol{a}^* = \min_{R^2, \boldsymbol{a}, \boldsymbol{\xi}} R^2 + \lambda \sum_{i=1}^{n} \xi_i, \tag{2.50}$$

$$||\boldsymbol{\varphi}(\boldsymbol{x}_i) - \boldsymbol{a}||^2 \leq R^2 + \xi_i \quad i = 1, \cdots, n, \tag{2.51}$$

$$\xi_i \geq 0 \quad i = 1, \cdots, n. \tag{2.52}$$

It is possible to re-write the problem with its Primal Lagrangian form

$$\mathscr{L}_p(R^2, \boldsymbol{a}, \boldsymbol{\xi}, \boldsymbol{\alpha}, \boldsymbol{\mu}) = R^2 + \lambda \sum_{i=1}^{n} \xi_i - \sum_{i=1}^{n} \alpha_i (R^2 + \xi_i - ||\boldsymbol{\varphi}(\boldsymbol{x}_i) - \boldsymbol{a}||^2) - \sum_{i=1}^{n} \mu_i \xi_i \tag{2.53}$$

$$\alpha_i, \mu_i \geq 0 \quad i = 1, \cdots n, \tag{2.54}$$

in order to find a solution also satisfying the bonds. So, similar to the SVM method, it is possible to define a Dual Lagrangian by computing the solutions of KKT conditions:

$$\frac{\delta \mathscr{L}}{\delta R^2} = 1 - \sum_{i=1}^{n} \alpha_i = 0, \tag{2.55}$$

$$\frac{\delta \mathscr{L}}{\delta \boldsymbol{a}} = -2 \sum_{i=1}^{n} (\boldsymbol{\varphi}(\boldsymbol{x}_i) - \boldsymbol{a}) \alpha_i = 0, \tag{2.56}$$

$$\frac{\delta \mathscr{L}}{\delta \xi_i} = \lambda - \alpha_i - \mu_i = 0. \tag{2.57}$$

The resulting Dual Lagrangian expression is:

$$\mathscr{L}_d(\boldsymbol{\alpha}) = \sum_{i=1}^{n} \alpha_i \boldsymbol{\varphi}(\boldsymbol{x}_i)^T \boldsymbol{\varphi}(\boldsymbol{x}_i) - \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j \boldsymbol{\varphi}(\boldsymbol{x}_i)^T \boldsymbol{\varphi}(\boldsymbol{x}_j), \tag{2.58}$$

which lead the following convex problem

$$\min_{\boldsymbol{\alpha}} \mathscr{L}_d(\boldsymbol{\alpha}) = \sum_{i=1}^{n} \alpha_i K(\boldsymbol{x}_i, \boldsymbol{x}_i) - \sum_{i=1}^{n}\sum_{j=1}^{n} \alpha_i \alpha_j K(\boldsymbol{x}_i, \boldsymbol{x}_j), \tag{2.59}$$

$$\sum_{i=1}^{n} \alpha_i = 1, \tag{2.60}$$

$$0 \leq \alpha_i \leq \lambda. \tag{2.61}$$

where $K(\boldsymbol{x}_i, \boldsymbol{x}_j) = \boldsymbol{\varphi}(\boldsymbol{x}_i))^T \boldsymbol{\varphi}(\boldsymbol{x}_j))$ is the considered Kernel function. It has to be noted that the problem can also be extended to non-linearly transformed spaces using the "Kernel trick" for distances as for SVM, SVR, and KRLS algorithms.

Finally, the result of this minimization process allows to find the best $\boldsymbol{\alpha}$ which, for the Representer's Theorem, define the function of the classifier.

$$f(\boldsymbol{x}) = \sum_{i=1}^{n} \alpha_i K(\boldsymbol{x}_i, \boldsymbol{x}). \tag{2.62}$$

The hyperparameters OCSVM $\mathcal{H}^{\text{OCSVM}}$ are the same as the ones of SVM.

2.6 Hybrid Models

Another down-side of pure DDMs is that they usually produce non-parametric models that are not supported by any physical interpretation; this may limit the interpretability of their results, since they do not exploit important knowledge about the phenomena of interest.

For this reason, HMs are used in order to combine physical knowledge of a phenomenon and statistical inference. In the latest years, models able to take advantage of the best characteristics of both PMs and DDMs by combining them together are becoming of much interest in many practical applications. HMs are widely used in those contexts were the experience on the field brought by PMs can enhance the DDMs prediction [COBA17].

Due to their mathematical formulation which easily allows including bonds and regularization terms, KMs can be easily adopted in order to define more complex models. In particular, KRLS can be used for building a model able to both take into account the physical knowledge about a phenomenon and the information hidden in the available data as the DDMs. This kind of approach is defined as HM, which takes the advantages of both PMs and DDMs. When targeting the development of a HM, it is ought to start from a simple observation: a HM, based on the previous observation, should be able to learn from the data without being too different, or far away, from the PM.

From the Data Science point of view, this requirement can be straightforwardly mapped in a typical ML Multi Task Learning (MTL) problem [Bax00, Car97, EP04, BH03, AEP08, Car97]. MTL aims at contemporary learning two concepts, in this case the PM and the available data, through a learning algorithm $\mathscr{A}_{\mathcal{H}}$ which exploits the data in $\mathcal{D}_n$ to learn a function f which is both close to the observation, the data $\mathcal{D}_n$ and the PM, namely its forecasts.

Consequently, in this case a slightly different scenario is presented where the dataset is composed by a triple of points $\mathcal{D}_n = \{(\boldsymbol{x}_1, y_1, p_1), \cdots, (\boldsymbol{x}_n, y_n, p_n)\}$ where p_i is the output of the PM in the point $\boldsymbol{x}_n$ with $i \in \{1, \cdots, n\}$. The target is to learn a function able to approximate both μ, namely the relation between the input $\boldsymbol{x} \in \mathcal{X}$ and the output $y \in \mathcal{Y}$, and the PM, namely the relation between the input and the output of the PM. Basically two tasks have to be learned. For this purpose there are two main approaches: the first approach is called Shared Task Learning (STL) and the second Independent Task Learning (ITL). While the latter independently learns a different model for each task, the former aims to learn a model that is common between all tasks. A well-known weakness of these methods is that they tend to generalize poorly on one of the two tasks [Bax00]. An appealing approach to overcome such limitations is provided by MTL [Bax00, Car97, EP04, BH03, AEP08]. This methodology leverages on the information between the tasks to learn more accurate models.

In order to apply the MTL approach to this case, it is possible to basically just modify the KRLS problem of Eq. (2.27) in order to contemporary learn a shared model and a task specific model which should be close to the shared model. In this way it is possible to obtain a model which is able to contemporary learn the two tasks. The interesting model is the first one while the task specific models are just used as a tool. A shared model is defined as

$$f(\boldsymbol{x}) = \boldsymbol{w}^T \boldsymbol{\varphi}(\boldsymbol{x}), \tag{2.63}$$

and two task specific models as

$$f_i(\boldsymbol{x}) = \boldsymbol{w}_i^T \boldsymbol{\varphi}(\boldsymbol{x}), \quad i \in \{y, p\}. \tag{2.64}$$

Then it is possible to state the MTL version of Eq. (2.27)

$$\begin{aligned} \boldsymbol{w}^*, \boldsymbol{w}_y^*, \boldsymbol{w}_p^* : \min_{\boldsymbol{w}, \boldsymbol{w}_y, \boldsymbol{w}_p} \quad & \sum_{i=1}^{n} \left[\boldsymbol{w}^T \boldsymbol{\varphi}(\boldsymbol{x}) - y_i\right]^2 + \left[\boldsymbol{w}^T \boldsymbol{\varphi}(\boldsymbol{x}) - p_i\right]^2 \\ & + \sum_{i=1}^{n} \left[\boldsymbol{w}_y^T \boldsymbol{\varphi}(\boldsymbol{x}) - y_i\right]^2 + \left[\boldsymbol{w}_p^T \boldsymbol{\varphi}(\boldsymbol{x}) - p_i\right]^2 \\ & + \lambda \|\boldsymbol{w}\|^2 + \theta(\|\boldsymbol{w} - \boldsymbol{w}_y\|^2 + \|\boldsymbol{w} - \boldsymbol{w}_p\|^2). \end{aligned} \tag{2.65}$$

where λ is the usual regularization of KRLS and $\theta \in [0, \infty)$, instead, is another hyperparameter that forces the shared model to be close to the task specific models. Basically the MTL problem of Eq. (2.65) is a concatenation of three learning problems solved with KRLS plus a term which tries to keep related all the three different problems.

By exploiting the kernel trick as in KRLS it is possible to reformulate Problem (2.65) as follows

$$\boldsymbol{\alpha}^* : \quad \min_{\boldsymbol{\alpha}} \quad \left\| \begin{bmatrix} Q & Q & 0 & 0 \\ Q & Q & 0 & 0 \\ 0 & 0 & Q & 0 \\ 0 & 0 & 0 & Q \end{bmatrix} \boldsymbol{\alpha} - \begin{bmatrix} \boldsymbol{y} \\ \boldsymbol{p} \\ \boldsymbol{y} \\ \boldsymbol{p} \end{bmatrix} \right\|^2 + \boldsymbol{\alpha}^T \begin{bmatrix} (\lambda+2\theta)Q & (\lambda+2\theta)Q & -\theta Q & -\theta Q \\ (\lambda+2\theta)Q & (\lambda+2\theta)Q & -\theta Q & -\theta Q \\ -\theta Q & -\theta Q & \theta Q & 0 \\ -\theta Q & -\theta Q & 0 & \theta Q \end{bmatrix} \boldsymbol{\alpha}, \tag{2.66}$$

where $\boldsymbol{p} = [p_1, \ldots, p_n]^T$. The solution of this problem is again equivalent to solving a simple linear system

$$\begin{bmatrix} Q+(\lambda+2\theta)I & Q+(\lambda+2\theta)I & -\theta I & -\theta I \\ Q+(\lambda+2\theta)I & Q+(\lambda+2\theta)I & -\theta I & -\theta I \\ -\theta I & -\theta I & Q+\theta I & 0 \\ -\theta I & -\theta I & 0 & Q+\theta I \end{bmatrix} \boldsymbol{\alpha}^* = \begin{bmatrix} \boldsymbol{y} \\ \boldsymbol{p} \\ \boldsymbol{y} \\ \boldsymbol{p} \end{bmatrix}. \tag{2.67}$$

The interesting function, the shared one, can be expressed as follows

$$f(\boldsymbol{x}) = \boldsymbol{w}^T \boldsymbol{\varphi}(\boldsymbol{x}) = \sum_{i=1}^{n} (\alpha_i + \alpha_{i+n}) K(\boldsymbol{x}_i, \boldsymbol{x}). \tag{2.68}$$

By exploiting the same FM and FS of the KRLS original formulation the HMs are obtained. What changes is the MS phase where λ, c, and p have been tuned as for the DDMs, but also θ.

Chapter 3

Development

During these studies, a lot of time and effort were dedicated to the development of DA methods for practical applications in the naval sector. In this chapter, the most relevant and innovative applications are presented, together with the main motivations which made them interesting use cases for DA methods. In particular, the proposed methods can be assigned to the following naval applications:

- Condition-Based Maintenance;
- Energy and Consumption Monitoring;
- System Safety;
- Components Design.

For each of the presented applications, a description of the particular field and use case is reported, together with a dataset description. Finally, for each application, the result of the adopted ML methods described in Section 2 are reported and discussed.

3.1 Condition-Based Maintenance of Naval Propulsion Systems: Data Analysis with Minimal Feedback.

The shipbuilding industry is particularly affected by maintenance problems, and much attention is invested every year in the analysis of marine applications. In fact, a ship breakdown necessarily requires a drydocking, and retrieving a stricken vessel offshore is not a trivial task [WY07, Mob02]. Recently, many ship builder companies are evaluating different DA solutions for improving the maintenance of their products, by monitoring the equipment exploiting the on-board network of sensors collecting data for security, diagnostic and monitoring purposes. DA can straightly take advantage of this technological substrate, by extracting, from the raw sensor data,

useful information about the efficiency of the ship, to identify the operational profiles, to reduce the fuel consumption, and to improve the maintenance activities. These data represent strategic information for shipyards, operators, ship owners, and crews, since they can be used for advisory, control, and fault detection purposes. Furthermore, DA allow exploiting exogenous data as well, such as weather information, which could contain some hidden information, potentially not easily representable with a conventional approach.

During this project, the problem of building effective DDMs to predict the main components decay state in a Naval Propulsion System (NPS) for CBM purposes was targeted. In particular, the decay of a vessel Gas Turbine (GT), Gas Turbine Compressor (GTC), Hull (HLL) and Propeller (PRP) is estimated. Examples of DA approaches applied to the marine industry can be found in [PBA+11], where a standard NN approach is used to improve monitoring of Gas Turbines, while Kernel based methods are applied in [SGK95], and [COG+16]. In [AAF08] and [BLS+13] image processing techniques are adopted for hull condition assessment. In [BU15] the engine and propeller state is predicted adopting an Artificial NN. A complete overview can be found in [LDMT16].

In particular, this project was first developed adopting a smaller amount of decayed NPS components and supervised ML regression models in order to predict their exact decay, as shown in [COG+16]. This work was extended in [COC+18b] in order to consider more than just two components at the same time [COG+16], by performing a regression analysis where the target is to estimate the actual decay state of the components described with an efficiency coefficient. Nevertheless, it was proven that a significant amount of historical data needed to be collected by experienced operators, together with the actual state of decay of each component. As a result, a different approach where collecting labelled samples is an easier task that can be performed by less experienced operators since the raw information about the decay is requested and it can be retrieved without impacting the ship activities [COC+18a]. For this reason, a traditional classification analysis was performed where the target is to estimate the label state of the components described with an efficiency coefficient. Some classification techniques were adopted to predict if the efficiency coefficient is above or below a certain threshold defined by the accepted loss in efficiency of the NPS components. In the end, also this approach resulted not feasible in a real-world scenario where the labelling process requires to stop the vessel or even to put the ship in a dry dock. Finally, the same problem has been tackled with another state-of-the-art approach which, in principle, does not need any labeled sample since it searches for novel behaviour in the data though a novelty detection algorithms [MS03, SMS+16]. Results show that with just a few labeled samples it is possible to fine tune this last methodology to achieve satisfying performances.

Specifically, three approaches are here reported and compared. Firstly, the results obtained with the regression models in order to predict the exact decay of the NPS decay values are reported [COC+18b]. Secondly, virtual sensors were built, able to continuously estimate the need for replacement of the components based on other sensors measurements which are indirectly

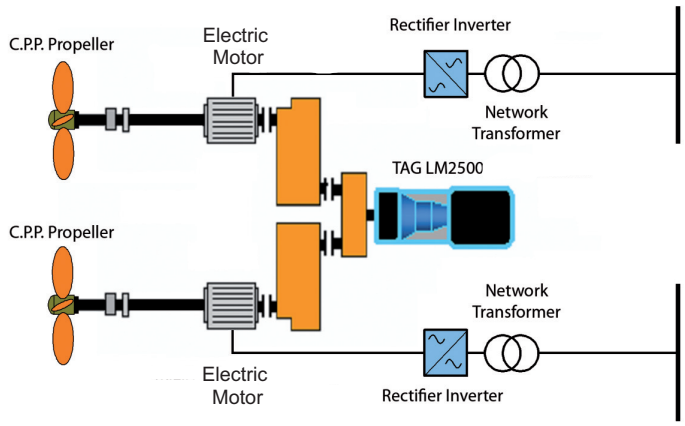


Figure 3.1: The CODLAG NPS setting.

influenced by this decay. Finally, the same analysis was performed, in conditions where only few labelled samples are present, by adopting a DDM which require a limited amount of information to achieve satisfying performance. The novelty of the proposed work relies on its ability of building a model whose accuracy is comparable with the state-of-the-art supervised learning techniques, adopting only an extremely limited number of labeled samples [COC+18a].

To this purpose, the analysis has been carried out comparing different state-of-the-art methodologies such as KMs [STC04], NNs [Ros58], GPs [Ras06], KNN [Duc00], and EMs [Bre01], reported in Chapter 2. In the following the data adopted for the experiments are described, together with the results of the three separate tests performed: the regression (REGR-PROB), the classification (CLASS-PROB), and the anomaly detection tasks (ANOMALY-PROB).

3.1.1 Available Data and Data Preprocessing

In this work, the focus was on a Frigate, characterised by a CODLAG NPS, widespread detailed in [COG+16]. In particular, the GT mechanically drives the two Controllable Pitch Propellers (CPP) through a cross-connected Gearbox (GB). Besides, each shaft has its Electric Propulsion Motor (EPM) mounted on the two shaft-lines. Two clutches between the GB and the two EPM and another clutch between the GT and the GB assure the possibility of using two different type of prime movers, i.e. EPM and GT. Finally, the electric power is provided by four Diesel Generators (DGs). The NPS setting is displayed in Figure 3.1.

This particular GB arrangement, allows the vessel to operate under different propulsive configurations to achieve the requirements of the vessel's mission profile. The vessel is characterised by the following mission profiles: Anti-Submarine Warfare (ASW), General-Purpose (GEP) and Anti-Aircraft Warfare (AAW). In particular, for the ASW profile, the EPMs are prime movers while the GT is disconnected through the clutches. Under the GEP mission profile, the GT is

the prime mover while the EPMs are working as shaft generators. Finally, for the AAW mission profile both the GT and the EPM are the prime movers. In this work, only the GT operating conditions have been taken into account.

In order to derive a proper dataset, an NPS numerical model developed in the Matlab® Simulink® software environment within many years of research [ABFC09] was considered. The numerical model is composed of several modules each one representing a single propulsion component such as the hull, the main engines, the propellers, the rudders, the GB, and the control system. In the previous literature, a model that considers the GT and GTC decay performance [COG$^+$16] was presented. The model was further improved to take into account the performance decay of the HLL and PRP, and can be used to undertake a holistic approach in addressing the performance decay by accounting the important components as follows:

1. Gas Turbine (GT);
2. Gas Turbine Compressor (GTC);
3. Hull (HLL);
4. Propeller (PRP).

The NPS model input parameters provided in [COC$^+$18b] are then reported as follows:

- Speed: this parameter is controlled by the control lever. The latter can only assume a finite number of positions lp_i with $i \in \{0, \cdots, 9\}$, which in turn correspond to a finite set of possible configurations for fuel flow and blade position. Each set point is designed to reach a desired speed v_i with $i \in \{0, \cdots, 9\}$:

$$v_i = 3 * lp_i \, [\text{Knots}], \quad \forall i \in \{0, \cdots, 9\}. \tag{3.1}$$

  Note that, if the transients is not taken into account, lp_i and v_i are deterministically related by a linear law. In the presented analysis the transients between different speeds have been not considered.
- The PRP thrust and torque decay limit over two years of operations are:

$$kK_t \in [0.9, 1.0], \quad kK_q \in [1.0, 1.1] \tag{3.2}$$

  kK_t and kK_q are respectively the components which define the decay of the torque and the thrust provided by the propeller in time. They are linearly correlated, since as the first decay of a certain quantity, the latter decay of the same quantity ($1 - kK_t = kK_q - 1$). For this reason only kK_t is analysed, considering the linear dependency between the two variables.
- The HLL decay has been modelled according to the available literature [HWB62]. The decay limits over two years of operations are:

$$kH \in [1, 1.2] \tag{3.3}$$

- The GT and GTC decay are

$$kM_c \in [0.95, 1.0], \quad kM_t \in [0.975, 1.0] \tag{3.4}$$

  which have been modelled considering the specific fuel consumption and the temperature of the exhaust gas increase due to fouling. The effect of the fouling is simulated by reducing the numerical values of the airflow rate M_c and of the isentropic efficiency η_c with the two reduction factors kM_c and kM_t.

The performance decay functions adopted have been empirically derived as functions of the time variable solely as described in [COC$^+$18b]. The real degradation behaviour of the physical asset should be defined through specific functions able to express the time dependency, the mutual interactions between the subsystems and the real operational profile. To overcome this issue, each possible combination of GTC, GT, HLL, and PRP decay status was considered based on the described functions, and sampled the range of decays with a uniform grid characterised by a degree of precision sufficient to have a proper granularity of representation. Given the above premises, the evolution of the system between two important dry dock maintenance for HLL and PRP can be exhaustively and realistically explored by simulating all its possible decayed states, as all the components are decaying the same time.

The space of possible states is described via the following tuple:

$$(lp, kK_t, kK_q, kH, kM_c, kM_t)_i, \quad i \in \{1, \cdots, 455625\} \tag{3.5}$$

since:

$$lp \in \mathcal{S}^{lp} = \{0, 3, 6, \cdots, 27\}, \tag{3.6}$$
$$kK_t \in \mathcal{S}^{kK_t} = \{0.9, 0.9 + {}^{0.1}/_{14}, 0.09 + {}^{0.2}/_{14}, \cdots, 1.0\}, \tag{3.7}$$
$$kK_q = 2 - kK_t, \tag{3.8}$$
$$kH \in \mathcal{S}^{kH} = \{1.0, 1.0 + {}^{0.2}/_{14}, 1.0 + {}^{0.4}/_{14}, \cdots, 1.2\}, \tag{3.9}$$
$$kM_c \in \mathcal{S}^{kM_c} = \{0.95, 0.95 + {}^{0.05}/_{14}, 0.95 + {}^{0.1}/_{14}, \cdots, 1.0\}, \tag{3.10}$$
$$kM_t \in \mathcal{S}^{kM_t} = \{0.975, 0.975 + {}^{0.025}/_{14}, 0.975 + {}^{0.05}/_{14}, \cdots, 1.0\}. \tag{3.11}$$

Note that the total number of samples 455625 is the result of making a simulation for each possible combination of decay status (15 values for GTC, 15 for GT, 15 for HLL, and 15 PRP) and speed (9 values). Once these quantities are fixed, the numerical model is run until the steady state is reached. Then, the model is able to provide all the quantities reported in Table 3.1. These subsets of models outputs are the same quantities that the automation system installed on-board can acquire and store.

The data generated with the reported procedure contain two sets of information: one regarding the quantities that the automation system installed on-board can acquire and store and the other

Table 3.1: Measured values available from the continuous monitoring system

#	Variable name	Unit	#	Variable name	Unit	#	Variable name	Unit
1	Lever (lp)	[ ]	10	Shaft speed (starboard)	[rpm]	19	TCS TIC control signal	[ ]
2	Vessel speed	[knots]	11	HP GT exit temperature	[^{o}C]	20	Thrust coefficient (starboard)	[ ]
3	GT shaft torque	[kN m]	12	Gas generator speed	[rpm]	21	PRP speed (starboard)	[rps]
4	GT speed	[rpm]	13	Fuel flow (mf)	[kg/s]	22	Thrust coefficient (port)	[ ]
5	PRP thrust (starboard)	[N]	14	TIC control signal	[%]	23	PRP speed (port)	[rps]
6	PRP thrust (port)	[N]	15	GTC outlet air pressure	[bar]	24	PRP torque (port)	[kN m]
7	Shaft torque (port)	[kN m]	16	GTC outlet air temperature	[^{o}C]	25	PRP torque (starboard)	[kN m]
8	Shaft speed (port)	[rpm]	17	External pressure	[bar]			
9	Shaft torque (starboard)	[kN m]	18	HP GT exit pressure	[bar]			

one regarding the associated state of decay (efficiency coefficient) of the different NPS components (GT, GTC, HLL, and PRP).

This problem can be straightforwardly mapped into a classical multi-output regression problem, as in [COG+16], where the aim is to predict the actual decay coefficient based on the automation data coming from the sensors installed on-board [Vap98]. Since in this analysis four different components of an NPS are taken into account (GT, GTC, HLL, and PRP), an incremental analysis was performed by breaking down the problem into simpler ones. In particular, first, only one NPS decayed component at the time is considered, then all the possible combination of two NPS decayed components are taken into account and so on until finally all the four NPS components are contemporarily considered. Consequently, since four NPS components are considered, $4 = \binom{4}{1}$ problems have to be solved when just one NPS decayed component at the time is considered, then $6 = \binom{4}{2}$ problems have to be solved when two NPS decayed components at the time are considered and so on. Finally, when all the four NPS components are contemporarily considered $1 = \binom{4}{4}$ problem has to be solved, for a total of $15 = \sum_{i=1}^{4} \binom{4}{i}$ problems. Therefore, from the dataset described, 15 sub-datasets corresponding to the cases mentioned above have been extracted. For the sake of clarity in Table 3.2 the 15 problems with the corresponding decay values are reported, not that in all the problems $lp \in \mathcal{S}^{lp}$. Unfourtunately, while the sensors' data coming from the automation system are easy to collect, the information regarding the associated state of decay is not so easy to retrieve. In fact, to circumvent this challenge to prove DDMs a numerical model for gathering all the information and build the dataset presented was exploited. In practice, instead, retrieving the state of decay of the different NPS components requires the intervention of an experienced operator and, in some cases, to stop the vessel or even to put the ship in a dry dock. Moreover, data-driven regression models require a huge amount of historical data and therefore a long acquisition time, which make them unfeasible in a real operational scenario.

Based on these considerations, it is possible to build simplified DDMs able to detect if the com-

Table 3.2: The 15 sub-problems corresponding to considering different decayed component at the time

Prob. #	Decayed Component Name	Non Decayed Component Name	Decayed Component	Non Decayed Component
1	GT	GTC,HLL,PRP	$kM_t \in \mathcal{S}^{kM_t}$	kM_c=1,kH=1,kK_t=1
2	GTC	GT,HLL,PRP	$kM_c \in \mathcal{S}^{kM_c}$	kM_t=1,kH=1,kK_t=1
3	HLL	GT,GTC,PRP	$kH \in \mathcal{S}^{kH}$	kM_c=1,kM_t=1,kK_t=1
4	PRP	GT,GTC,HLL	$kK_t \in \mathcal{S}^{kK_t}$	kM_c=1,kM_t=1,kH=1
5	GT,GTC	HLL,PRP	$kM_t \in \mathcal{S}^{kM_t}$,$kM_c \in \mathcal{S}^{kM_c}$	kH=1,kK_t=1
6	GT,HLL	GTC,PRP	$kM_t \in \mathcal{S}^{kM_t}$,$kH \in \mathcal{S}^{kH}$	kM_c=1,kK_t=1
7	GT,PRP	GTC,HLL	$kM_t \in \mathcal{S}^{kM_t}$,$kK_t \in \mathcal{S}^{kK_t}$	kM_c=1,kH=1
8	GTC,HLL	GT,PRP	$kM_c \in \mathcal{S}^{kM_c}$,$kH \in \mathcal{S}^{kH}$	kM_t=1,kK_t=1
9	GTC,PRP	GT,HLL	$kM_c \in \mathcal{S}^{kM_c}$,$kK_t \in \mathcal{S}^{kK_t}$	kM_t=1,kH=1
10	HLL,PRP	GT,GTC	$kH \in \mathcal{S}^{kH}$,$kK_t \in \mathcal{S}^{kK_t}$	kM_t=1,kM_c=1
11	GT,GTC,HLL	PRP	$kM_t \in \mathcal{S}^{kM_t}$,$kM_c \in \mathcal{S}^{kM_c}$,$kH \in \mathcal{S}^{kH}$	kK_t=1
12	GT,GTC,PRP	HLL	$kM_t \in \mathcal{S}^{kM_t}$,$kM_c \in \mathcal{S}^{kM_c}$,$kK_t \in \mathcal{S}^{kK_t}$	kH=1
13	GT,HLL,PRP	GTC	$kM_t \in \mathcal{S}^{kM_t}$,$kH \in \mathcal{S}^{kH}$,$kK_t \in \mathcal{S}^{kK_t}$	kM_c=1
14	GTC,HLL,PRP	GT	$kM_c \in \mathcal{S}^{kM_c}$,$kH \in \mathcal{S}^{kH}$,$kK_t \in \mathcal{S}^{kK_t}$	kM_t=1
15	GT,GTC,HLL,PRP		$kM_t \in \mathcal{S}^{kM_t}$,$kM_c \in \mathcal{S}^{kM_c}$,$kH \in \mathcal{S}^{kH}$,$kK_t \in \mathcal{S}^{kK_t}$	

ponent state of decay is above or below a certain threshold. These thresholds represent the accepted loss in efficiency of the NPS components and the consequently sustainable costs of keeping a less performing vessel operative. This approach represents an abstraction of the problem which allows a more practical collection of the state of decay of the component. In fact, instead of requiring the precise state of decay, this approach only requires detecting if the decay state of the components is acceptable or not. Consequently, the collection of these data can be performed by less experienced operators since raw information about the decay is requested and can be retrieved without impacting the ship activities. This new problem can then be mapped into a multi-output binary classification problem [Vap98] where the aim is to predict if the decay state of an NPS component is acceptable or not based on the automation data coming from the sensors installed on-board.

The data previously described can be easily exploited to tackle this new problem as well. In fact, by thresholding kK_t, kH, kM_c, and kM_t the corresponding binary valued state of decay of the NPS components are obtained. In other words, if the efficiency coefficients are above or below a defined threshold, based on the accepted loss in efficiency of the NPS components, they are tagged as "decayed" or "not decayed". Thresholds were fixed according to the least affordable value of decay of the single component. Defining these thresholds is not a trivial task. The proposed approach is to define the maximum level of inefficiency that the operator or the shipowner is willing to tolerate before taking action and re-establish the efficiency of the system.

Two years is considered as a typical time frame between two important dry dock maintenance for HLL and PRP.

The HLL and PRP thresholds have been defined considering one year of operation. The proposed limits are just an example of the possible selection that is possible to set up in order to implement a CBM framework

$$kK_t \begin{cases} [0.9 - 0.95) & \text{decayed} \\ [0.95 - 1] & \text{not decayed} \end{cases} \tag{3.12}$$

$$kH \begin{cases} (1.1 - 1.2] & \text{decayed} \\ [1 - 1.1] & \text{not decayed} \end{cases} \tag{3.13}$$

As for GT and GTC, an effective time service of 2000 hours per year is considered as a reasonable operating time for these vessel types. In agreement with these observations the following thresholds can be defined based on the knowledge of the time domain decay functions:

$$kM_c \begin{cases} [0.95 - 0.98) & \text{decayed} \\ [0.98 - 1] & \text{not decayed} \end{cases} \tag{3.14}$$

$$kM_t \begin{cases} [0.975 - 0.99) & \text{decayed} \\ [0.99 - 1] & \text{not decayed} \end{cases} \tag{3.15}$$

Unfortunately, results show that estimating if the decay state is acceptable or not, instead of estimating its specific state, remarkably reduces the number of samples required to find accurate DDMs. However, this quantity is still too large with respect to what can be collected in a real operational scenario.

To solve this issue, its is possible to look at the same problem from another perspective. Specifically, it is reasonable to state that, for the vast majority of the time, NPS components of the ships operate in an acceptable state of decay. Consequently, most of the sensor data collected by the automation system represent ordinary operating conditions corresponding to a reasonable decay state of the NPS components (GT, GTC, HLL, and PRP). Just very few times during the ship lifetime it happens that it has to operate with over-decayed components. If, for some reasons, one or more NPS components decay too fast, the corresponding automation data measurements deviate from their expected behaviour. This new problem can be straightforwardly mapped into a classical outlier (novelty) detection problem [Haw80, STC04, SMS$^+$16] where the aim is to detect unexpected behaviour in the sensor data collected by the automation system which may correspond to an over-decayed state of an NPS components. This method does not require to know either the actual state of decay of the components, as a regression task would do, or the less detailed information about "decayed" or "not decayed", as in the binary classification framework. In this case, the method just needs the sensor data collected by the automation system (see

Table 3.1) without any supervision or feedback from the operator. These kinds of DDMs try to build a model of the "usual" operational profile of the ship and automatically detect if the sensor data collected by the automation system are "deviating too much" from the established behaviour. In this context "usual" means that the efficiency of GT, GTC, HLL, and PRP are in the acceptable range while "deviating too much" means that they are not in the acceptable range, according to Eqns. (3.12), (3.13), (3.14), and (3.15).

As for the binary classification framework, the data described can be easily exploited to tackle this problem as well. In fact, it is just necessary to keep the data corresponding to an acceptable decay state with respect to kK_t, kH, kM_c, and kM_t and in accordance with Eqns. (3.12), (3.13), (3.14), and (3.15). Finally, for testing and tuning the DDMs, it is possible to use just a few samples of the dataset corresponding to an unacceptable decay state. Note that these are the only samples which are costly to retrieve since they are the only ones that require the intervention of expert operators. Results show that with just very few samples (≈ 10) of decayed state of the vessel, it is possible to obtain effective DDMs for CBM of NPS.

As a final remark, it is possible to recall that each navy frigate is characterised by different mission profiles (AAW, ASW, and GEP). Each mission profile is characterised by a particular use of the ship in terms of speed. CBM DDMs for NPS do not need to estimate the state of decay of the NPS components for all the possible mission profiles. In fact, the vessel operates at a cruise speed (which is approximately ≈ 15 knots) while the time spent by the vessel at different speeds is negligible. For this reason, the same analysis described in the previous paragraph was conducted by setting $lp = 15$. Results show that this simplification will further reduce the amount of historical data needed to build effective CBM DDMs for NPS.

3.1.2 Regression Results

In this section, the results obtained by different regression methods proposed in 2 applied to the CBM of the main components of an NPS, in the proposed regression framework are reported.

As described in 3.1.1, the regression problem was the first to be tackled where the actual value of the decay parameters needs to be estimated. The regression techniques which were considered to perform this analysis were DNNs, SNNs and SELM reported in Section 2.4.1, SVR, and KRLS with Gaussian Kernel reviewed in Section 2.4.2, RF and RRE described in Section 2.4.3, and finally GP and KNN Section 2.4. The different datasets considered, corresponding to the 15 problems of Table 3.2 $\{P1, \cdots, P15\}$, were divided into training and test set, respectively $\mathcal{D}_n$ and $\mathcal{T}_m$, as reported in Chapter 2. Moreover, different dimensions of the training set $n \in \{10, 24, 55, 130, 307, 722, 1700, 4000\}$ were considered. For each supervised regression learning task and each of the following ML technique, the BTS MS procedure was performed with $r = 1000$, as described in Section 2.2. Here-below, the list of hyperparameters tested during the MS, with their respective intervals, is reported:

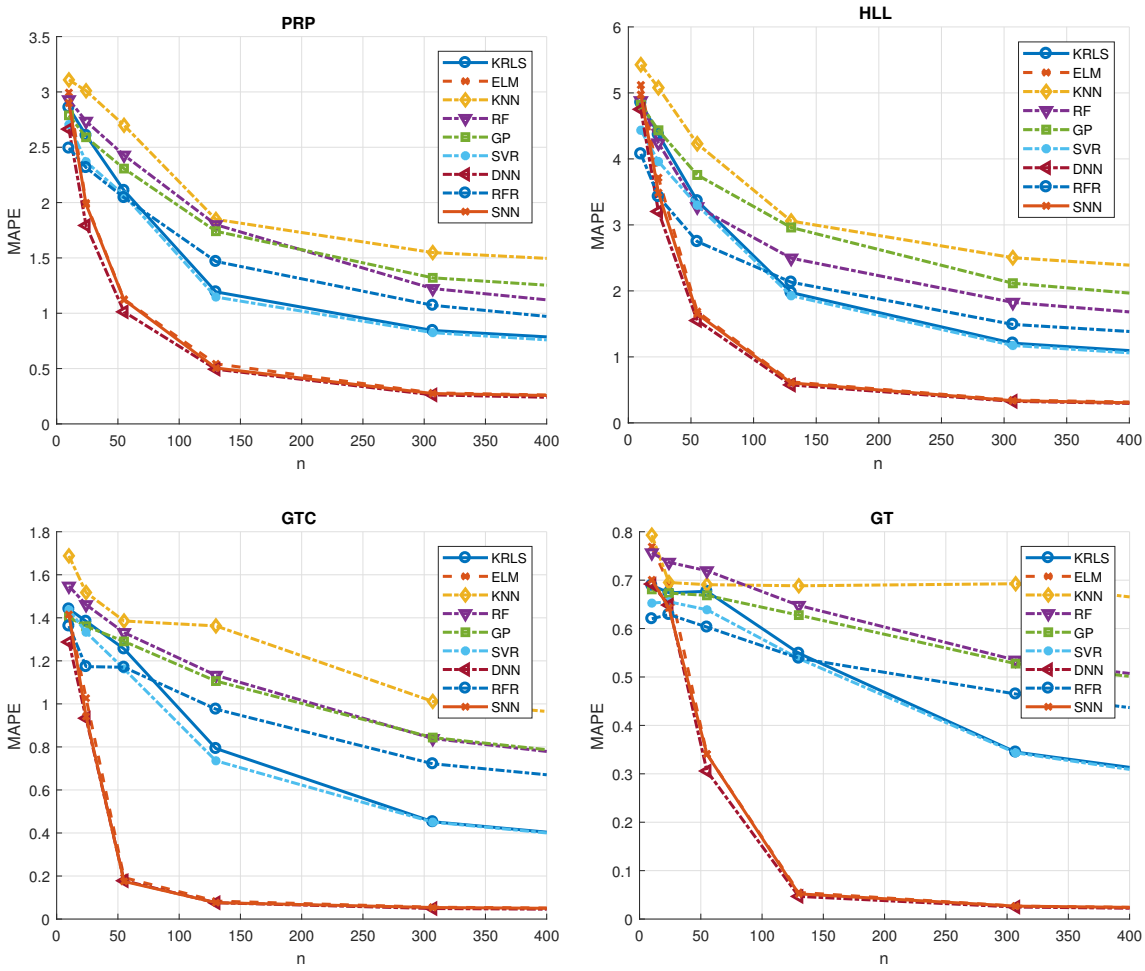


Figure 3.2: REGR-PROB: MAPE of the models learned with the different algorithms (DNN, SNN, ELM, SVR, KRLS, KNN, and GP) when varying n for $P15$ and for the four main NPS components.

1. DNN: the set of hyperparameters is $\mathcal{H}^{\text{DNN}} = \{h_1, h_{2,1}, \cdots, h_{2,h_1}\}$ in $\mathfrak{H}^{\text{DNN}} = \{1, 3, 5, 7, 10\} \times \{10, 10^{1.2} \cdots, 10^3\} \times \cdots \times \{10, 10^{1.2} \cdots, 10^3\}$;
2. SNN: the set of hyperparameters is $\mathcal{H}^{\text{SNN}} = \{h_1\}$ in $\mathfrak{H}^{\text{SNN}} = \{1, 3, 5, 7, 10\}$;
3. ELM: the set of hyperparameters is $\mathcal{H}^{\text{ELM}} = \{h_1, h_2\}$ in $\mathfrak{H}^{\text{ELM}} = \{10, 10^{1.2}, \cdots, 10^3\} \times \{10^{-2}, 10^{-1.5} \cdots, 10^2\}$;
4. SVR: the set of hyperparameters is $\mathcal{H}^{\text{SVR}} = \{h_1, h_2, h_3\}$ in $\mathfrak{H}^{\text{SVR}} = \{10^{-6}, 10^{-5}, \cdots, 10^3\} \times \{10^{-2}, 10^{-1.4}, \cdots, 10^3\} \times \{10^{-2}, 10^{-1.4}, \cdots, 10^3\}$;
5. KRLS: the set of hyperparameters is $\mathcal{H}^{\text{KRLS}} = \{h_1, h_2\}$ in $\mathfrak{H}^{\text{KRLS}} = \{10^{-2}, 10^{-1.4}, \cdots, 10^3\} \times \{10^{-2}, 10^{-1.4}, \cdots, 10^3\}$;
6. KNN: the set of hyperparameters is $\mathcal{H}^{\text{KNN}} = \{h_1\}$ in $\mathfrak{H}^{\text{KNN}} = \{1, 3, 7, 13, 27, 51\}$;
7. GP: the set of hyperparameters is $\mathcal{H}^{\text{GP}} = \{h_1\}$ in $\mathfrak{H}^{\text{GP}} = \{10^0, 10^{0.3}, \cdots, 10^3\}$;

The performances of each model are measured according to the MAPE metric described in Section 2.1. Each experiment was performed 10 times in order to obtain statistical relevant result.

For SNN and DNN the Python Keras library [1] has been exploited. For ELM, SVR, KRLS, and KNN, a custom R implementation has been developed. For RF the R package of [LW02] has been exploited. For RRE the implementation of [BF15] has been exploited. For GP the R package of [ZHSK04] has been exploited.

Note that, based on the problem under exam, just a subset of the components may decay. Moreover, since this dataset cardinality is limited, it is possible that a particular combination of problem and cardinality of the training set cannot be tested. For example, when just one component decays, nearly one hundred samples are available (see Section 3.1) and then a maximum of $n = 55$. Note that, in a real-world scenario, the only useful problem is $P15$, namely when all the components decay contemporarily. In Figures 3.2 the MAPE of the models is reported learned with the different algorithms when varying n for $P15$ and the four NPS components. In Figures 3.3 the MAPE of the DNN (the best performing model) is reported when varying n for the different problems under examination and the four NPS components.

From the different figures it is possible to observe that:

- as expected the larger is n the better performances are achieved by the learned models (see Figures 3.3 and 3.2);
- the models learned with ELM, SNN, and especially DNN generally show the best performances (see Figure 3.2);
- the larger is the number of decaying components, the lower performances are achieved by the learned models (see Figure 3.3);
- the most complicate decay to predict is the one of the HLL, in fact the problems where the HLL decays are the ones which show lower accuracies (see Figure 3.3);
- unfortunately, as it can be seen from Figures 3.3 and 3.2, this approach requires a significant amount of labelled data, which could not be retrieved in many operational scenarios.

[1] `https://github.com/fchollet/keras`

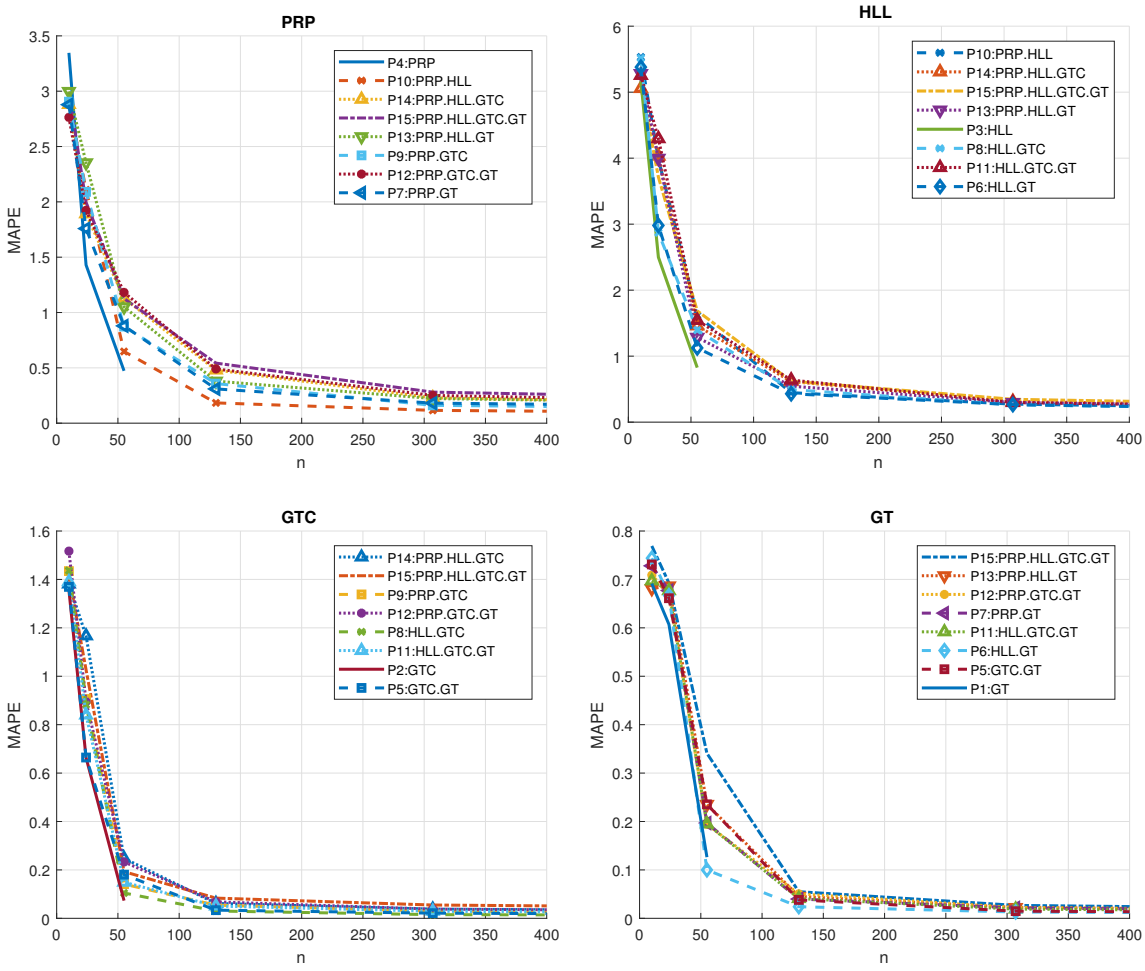


Figure 3.3: REGR-PROB: MAPE of the models learned with DNN when varying n for the different problems $P1, \cdots, P15$ and for the four main NPS components.

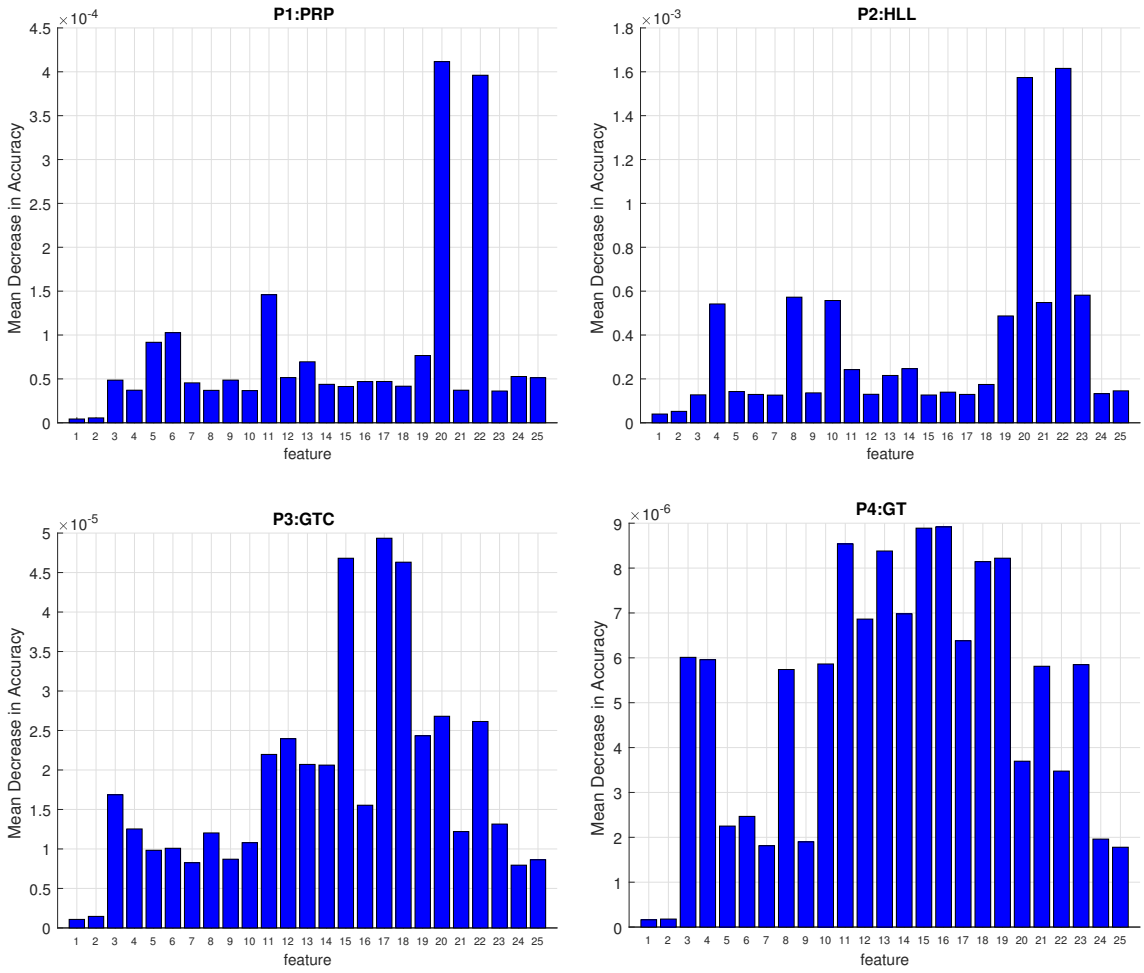


Figure 3.4: REGR-PROB: FS performed with RF for the four main NPS components for problem $P1$, $P2$, $P3$, and $P4$

> In fact, while the sensors data coming from the automation system are easy to collect, the information regarding the associated state of decay is not so easy to retrieve. Collecting the state of decay of the different NPS components requires the intervention of an experienced operator and, in some cases, to stop the vessel or even to put the ship in a dry dock.

Finally in Figure 3.4 and 3.5 the FS phase is reported, performed with the RF procedure described in 2.4.3, for $P1$, $P2$, $P3$, and $P4$ (see Figure 3.4) and for $P15$ (see Figure 3.5). In particular, for each problem and each feature, the mean decrease in accuracy is reported as described in Chapter 2.

From Figure 3.4 and 3.5 it is possible to observe that:

- as expected, when just one component decays (see Figure 3.4), few predictors have strong

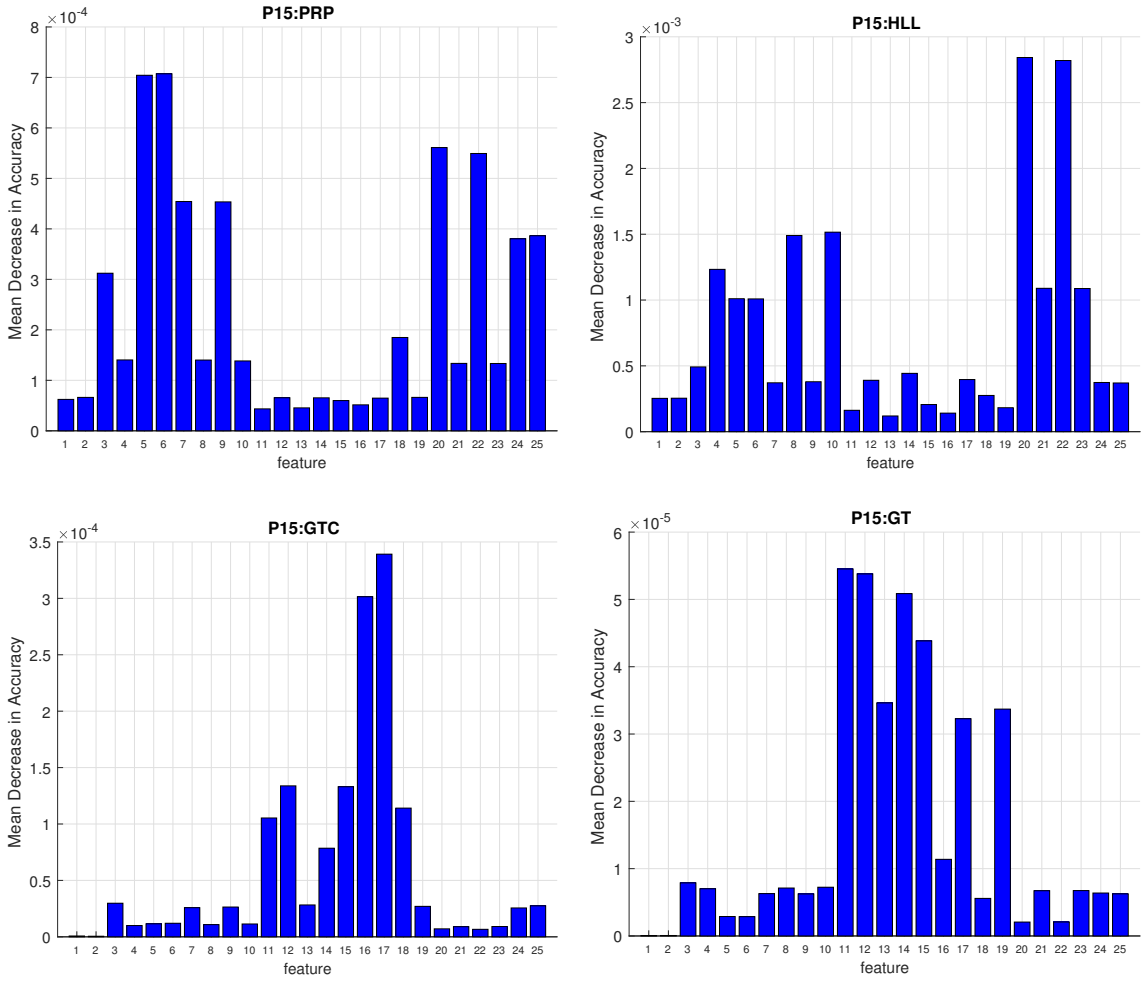


Figure 3.5: REGR-PROB: FS performed with RF for the four main NPS components for problem $P15$.

predictive power while when all the components decay (see Figure 3.5) many predictors need to be considered in order to achieve satisfying accuracies;

- from Figure 3.4, it is possible to note that the RF model can adequately account for the relevant features as the outcome is consistent with the available knowledge (note that even if DNN has higher predictive capabilities with respect to RF the latter is still competitive, see Figure 3.2). In fact, for the $P1$ and $P2$ (PRP and HLL decays) the features 22 and 24 (Thrust coefficient stbd and port) have strong predictive power. Moreover, for the $P3$ (GTC decay) the features describing the thermodynamic process, $17, 19$, and 20 (GT Compressor outlet air pressure, External Pressure ,and HP Turbine exit pressure) have the highest predictive power. Finally, several features are necessary for the GT decay prediction in $P4$.

3.1.3 Classification Results

In this section, the results obtained by different classification methods proposed in 2 applied to the CBM of the main components of an NPS, in the proposed classification framework are reported. Since in Section 3.1.2 it was proved that the larger is the number of decaying components, the lower performances are achieved by the learned models, only the model considering all the four components decaying at the same time is considered, in order to put the analysis in the more realistic scenario as possible. The dataset considered was divided into training and test set, as in Section 3.1.2. The classification techniques which were considered to perform this analysis were DNNs, SNNs and SELM reported in Section 2.4.1, SVM, and KRLS with Gaussian Kernel reviewed in Section 2.4.2, RF and RRE described in Section 2.4.3, and finally GP and KNN in Section 2.4. Here-below, the list of hyperparameters tested during the MS, with their respective intervals, is reported:

1. DNN: the set of hyperparameters is $\mathcal{H}^{\text{DNN}} = \{h_1, h_{2,1}, \cdots, h_{2,h_1}\}$ in $\mathfrak{H}^{\text{DNN}} = \{1, 3, 5, 7, 10\} \times \{10, 10^{1.2} \cdots, 10^3\} \times \cdots \times \{10, 10^{1.2} \cdots, 10^3\}$;
2. SNN: the set of hyperparameters is $\mathcal{H}^{\text{SNN}} = \{h_1\}$ in $\mathfrak{H}^{\text{SNN}} = \{1, 3, 5, 7, 10\}$;
3. ELM: the set of hyperparameters is $\mathcal{H}^{\text{ELM}} = \{h_1, h_2\}$ in $\mathfrak{H}^{\text{ELM}} = \{10, 10^{1.2}, \cdots, 10^3\} \times \{10^{-2}, 10^{-1.5} \cdots, 10^2\}$;
4. SVM: the set of hyperparameters is $\mathcal{H}^{\text{SVM}} = \{h_1, h_2\}$ in $\mathfrak{H}^{\text{SVM}} = \{10^{-2}, 10^{-1.4}, \cdots, 10^3\} \times \{10^{-2}, 10^{-1.4} \cdots, 10^3\}$;
5. KRLS: the set of hyperparameters is $\mathcal{H}^{\text{KRLS}} = \{h_1, h_2\}$ in $\mathfrak{H}^{\text{KRLS}} = \{10^{-2}, 10^{-1.4}, \cdots, 10^3\} \times \{10^{-2}, 10^{-1.4}, \cdots, 10^3\}$;
6. KNN: the set of hyperparameters is $\mathcal{H}^{\text{KNN}} = \{h_1\}$ in $\mathfrak{H}^{\text{KNN}} = \{1, 3, 7, 13, 27, 51\}$;
7. GP: the set of hyperparameters is $\mathcal{H}^{\text{GP}} = \{h_1\}$ in $\mathfrak{H}^{\text{GP}} = \{10^0, 10^{0.3}, \cdots, 10^3\}$;

As for REGR-PROB, when RF is exploited, also the FS phase is performed to understand how the DDM combines the different features in order to predict the decay state of each component.

The classification models implementation have been performed adopting the libraries reported in the REG-PROB, for their regression counterparts were possible.

In Figures 3.6 the AMR of the models learned with the different algorithms is reported, when varying n and for the four main NPS components. In Figures 3.7 the AMR of the DNN (the best performing model) is reported, when varying n and for the four main NPS components.

From the different tables and figures it is possible to observe that:

- the larger is n the better performances are achieved by the learned models (see Figure 3.6) and the models learned with ELM, SNN, and especially DNN generally show the best performances (see Figure 3.6);
- as expected, to achieve a reasonable AMR a smaller number of samples is needed with respect to a regression-based approach not feasible in practice.

In Figure 3.8 the FS phase for the four main NPS components is reported. Taking into account the problem $P15$, for each feature reported in Table 3.1, the mean decrease in accuracy as described in Chapter 2 is reported. From Figure 3.8, it is possible to note that the RF model can adequately account for the relevant features as the outcome is consistent with the available knowledge [COC$^+$18b]. According to Figure 3.8, several features are always necessary to forecast the decay state of each component. As far as the PRP component is concerned, the thrust features (5 and 6), the shaft torque features (7 and 9) and the PRP torque features (24 and 25) have high predictive power. As expected, for the HLL component the thrust coefficients features (20 and 22) have the most significant predictive power. When it comes to the GTC component, the features describing the thermodynamic process have the highest predictive power, nominally GTC outlet air temperature, External pressure, and HP GT exit temperature features (16, 17 and 11). Finally, for the GT component prediction, several features are necessary, also this case is in line with engineering state-of-the-art knowledge [COC$^+$18b]. These results indicate that, from a data driven perspective, the decay state of each component influences different phases of the NPS behavior.

Figure 3.8 clearly shows that the interaction between the main components cannot be easily modelled with a physical approach, considering the large number of variables that affect the final behaviour of each component. Instead, DDMs, by making use of these variables, can outperform PMs as they have the capability to take into account all the available sensors measurements to build effective and accurate predictors as reported in Figure 3.6.

3.1.4 Anomaly Detection Results

In this section, the results obtained by different anomaly detection methods proposed in Chapter 2 applied to the CBM of the main components of an NPS are reported. The anomaly detection techniques which were considered to perform this analysis were OCSVM and GKNN, reported in

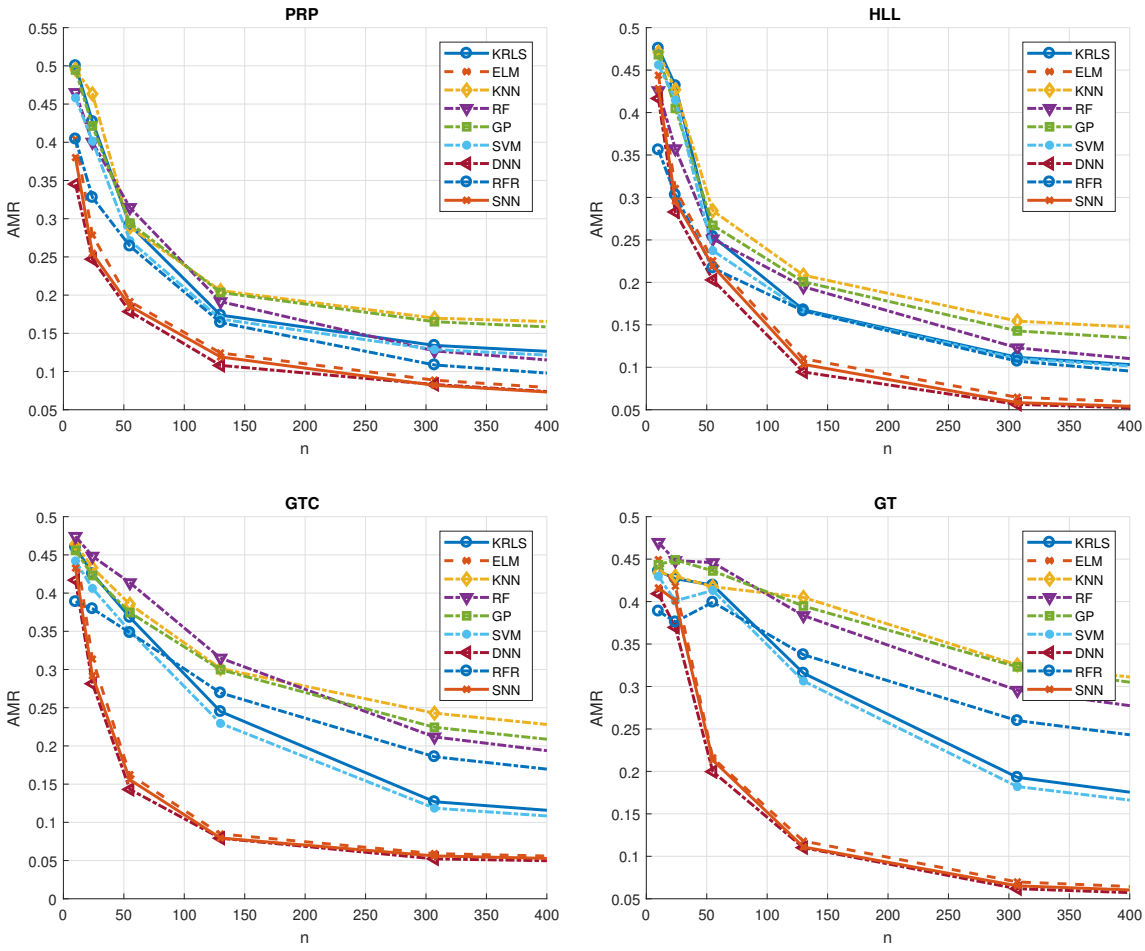


Figure 3.6: CLASS-PROB: AMR of the models learned with the different algorithms when varying n and for the four main NPS components.

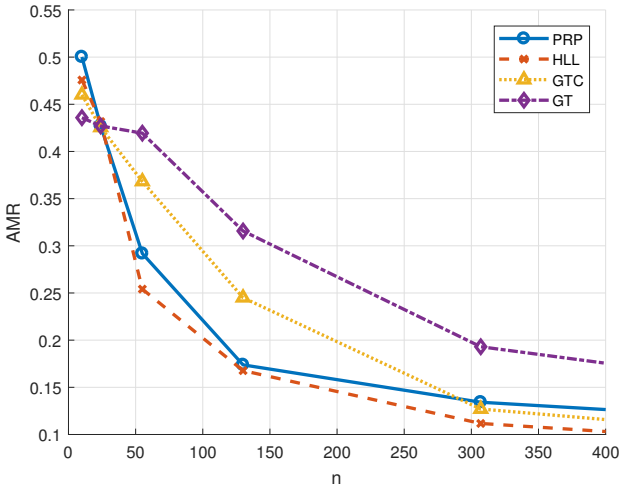


Figure 3.7: CLASS-PROB: AMR of the models learned with DNN when varying n for the four main NPS components.

Section 2.5. Similarly to the supervised learning task, in the unsupervised case different dimensions of the training set were considered $n \in \{1500, 2000, 3000, 4000\}$ and the MS procedure was performed as follows:

1. OCSVM: the set of hyperparameters is $\mathcal{H}^{\text{OCSVM}}=\{h_1, h_2\}$ in $\mathfrak{H}^{\text{OCSVM}}=\{10^{-4}, 10^{-3.7}, \cdots, 10^{3}\} \times \{10^{-4}, 10^{-3.8}, \cdots, 10^{-1.0}\}$;
2. GKNN: the set of hyperparameters is $\mathcal{H}^{\text{GKNN}}=\{h_1\}$ in $\mathfrak{H}^{\text{GKNN}}=\{1, 3, 7, 13, 27, 51\}$;

The $\mathcal{V}_v^r$ cardinality was varied $v \in \{10, 20, 30\}$ with linear step, in order to test the possibility of building an efficient model with a few labeled samples. Note that, also in this case, the BTS MS procedure is adopted with $r = 1000$ and that the labels are only needed in $\mathcal{V}_v^r$ and not in $\mathcal{L}_l^r$ as described in Section 2.2. Each experiment was performed 10 times in order to obtain statistical relevant result, and the t-student 95% confidence interval is reported when space in the table was available without compromising their readability.

In Table 3.3, for PRP, HLL, GTC, and GT, the AMR of the models learned with the different algorithms (OCSVM and GKNN) is reported, when the number of unlabelled samples in the learning set is $l = 4000$ and when varying the number of labeled samples in the validation set $v \in \{10, 20, 30\}$ (half positively and half negatively labeled). In Table 3.4, respectively for PRP, HLL, GTC, and GT, the AMR of the models learned with the different algorithms is reported, when $v = 30$ and when $l \in \{1500, 2000, 3000, 4000\}$. In Table 3.5, for PRP, HLL, GTC, and GT, the different indexes of performances (AMR, TP, TN, FP, and FN) of the models learned with the different algorithms are reported when $n = 4000$ and $v = 30$.

From the tables it is possible to observe that:

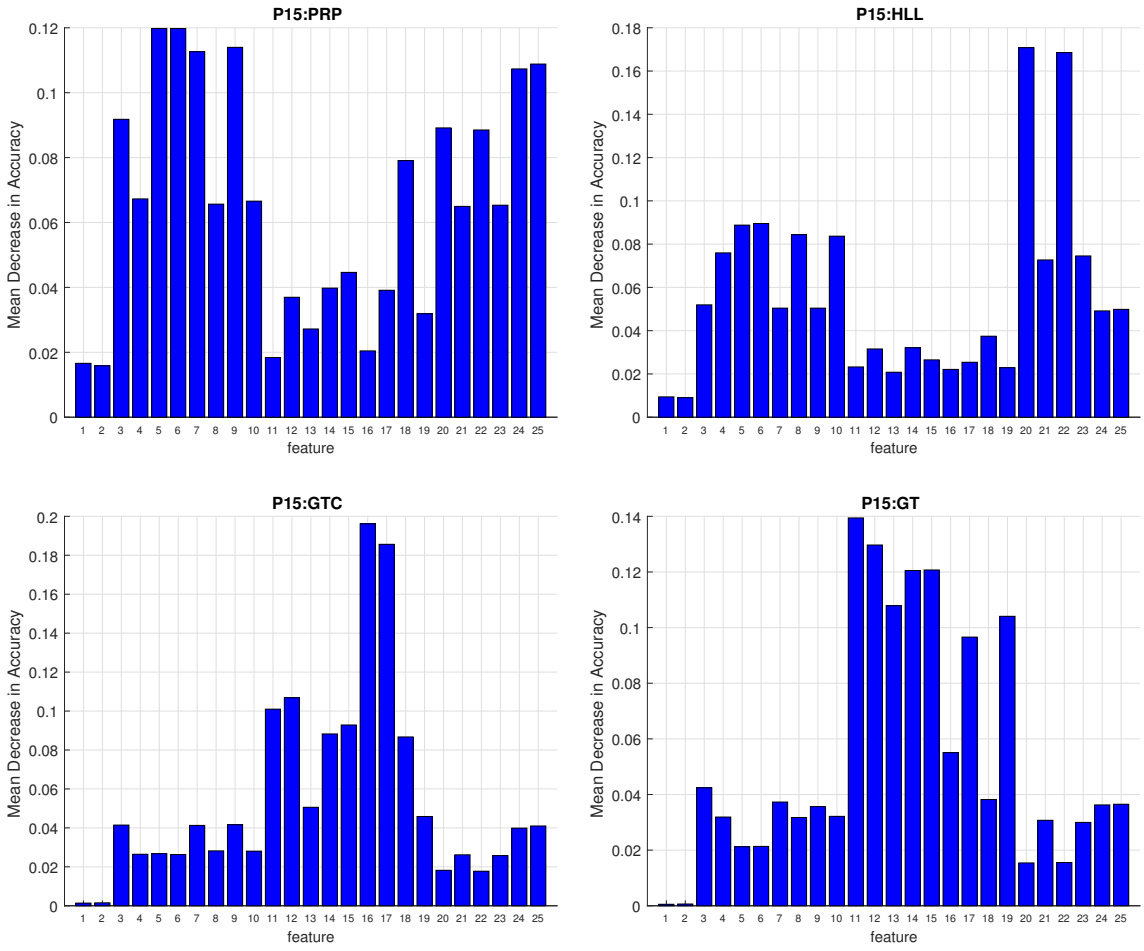


Figure 3.8: CLASS-PROB: FS performed with RF for the four main NPS components.

Table 3.3: ANOMALY-PROB: AMR of the models learned with the different algorithms (OCSVM and GKNN) when $l = 4000$ and $v \in \{10, 20, 30\}$ and for the four main NPS components.

	v	PRP	HLL	GTC	GT
OCSVM	10	0.08 ± 0.08	0.07 ± 0.09	0.05 ± 0.07	0.11 ± 0.06
	20	0.08 ± 0.10	0.08 ± 0.07	0.10 ± 0.07	0.09 ± 0.06
	30	0.08 ± 0.07	0.12 ± 0.08	0.10 ± 0.07	0.09 ± 0.03
GKNN	10	0.07 ± 0.07	0.07 ± 0.09	0.04 ± 0.07	0.10 ± 0.06
	20	0.08 ± 0.10	0.08 ± 0.07	0.08 ± 0.07	0.07 ± 0.06
	30	0.08 ± 0.07	0.12 ± 0.08	0.08 ± 0.08	0.08 ± 0.03

Table 3.4: ANOMALY-PROB: AMR of the models learned with the different algorithms (OCSVM and GKNN) when $l \in \{1500, 2000, 3000, 4000\}$ and $v = 30$ and for the four main NPS components.

	n	PRP	HLL	GTC	GT
OCSVM	1500	0.09 ± 0.04	0.14 ± 0.09	0.11 ± 0.09	0.08 ± 0.03
	2000	0.12 ± 0.10	0.11 ± 0.04	0.10 ± 0.04	0.11 ± 0.13
	3000	0.08 ± 0.06	0.11 ± 0.04	0.13 ± 0.17	0.08 ± 0.05
	4000	0.08 ± 0.07	0.12 ± 0.08	0.10 ± 0.07	0.09 ± 0.03
GKNN	1500	0.09 ± 0.04	0.12 ± 0.09	0.10 ± 0.09	0.06 ± 0.03
	2000	0.11 ± 0.11	0.12 ± 0.04	0.08 ± 0.04	0.08 ± 0.13
	3000	0.07 ± 0.06	0.10 ± 0.04	0.12 ± 0.16	0.09 ± 0.04
	4000	0.08 ± 0.07	0.12 ± 0.08	0.08 ± 0.08	0.08 ± 0.03

- both OCSVM and GKNN perform quite well on the problem and there is no clear winner;
- changing l or v does not remarkably affect the performance of the models;
- with just few labeled samples, around 10, it is possible to obtain satisfying accuracies and this is quite a remarkable result, since 10 samples can be easily manually labeled by an expert operator;
- in some cases, the mean value of the reported AMR increases instead of decreasing when the number of training samples increases; as a fact, this value is subject to statistical variation in the data whose variance in the results can be considered acceptable from a statistical point of view. Note also that the number of validation samples is very limited and this high variance is also justified by this fact;
- FP and FN rate are quite balanced and this is a further indication of the quality of the result.

Table 3.5: ANOMALY-PROB: the different indexes of performances (AMR, TP, TN, FP, and FN) of the models learned with the different algorithms (OCSVM and GKNN) when $n = 4000$ and $v = 3$ and for the four main NPS components.

		PRP	HLL	GTC	GT
OCSVM	AMR	0.08±0.07	0.12±0.08	0.10±0.07	0.09±0.03
	TP	46.68±8.88	45.29±9.25	47.89±3.17	46.59±5.49
	TN	45.47±3.88	42.56±7.54	42.42±9.36	44.56±4.50
	FN	3.32±8.88	4.71±9.25	2.11±3.17	3.41±5.49
	FP	4.53±3.88	7.44±7.54	7.58±9.36	5.44±4.50
GKNN	AMR	0.08±0.07	0.12±0.08	0.08±0.08	0.08±0.03
	TP	46.68±8.66	45.89±9.04	48.04±3.16	47.42±5.41
	TN	45.24±3.86	42.30±7.61	44.14±9.55	44.49±4.47
	FN	3.32±8.84	4.11±9.21	1.96±3.13	2.58±5.54
	FP	4.76±3.87	7.70±7.54	5.86±9.29	5.51±4.56

3.2 Unintrusive Monitoring of Induction Motors Bearings via Deep Learning on Stator Currents.

Induction Motors (IMs) are ubiquitous in many industrial systems, such as modern automation systems, e-cars, and ships [BGRR08, TF01, DAK15]. In particular, IMs are cheap, characterized by a reasonably high efficiency, and require low maintenance activities [KCMS16]. However, IMs are subject to different types of undesirable faults which cause additional costs and losses in production time [Dek96]. Moreover, decayed components inside the IMs often result in a higher power consumption with respect to properly maintained ones, thus requiring additional costs in energy supply [Tsa02]. As a result, IMs maintenance is a critical problem which requires the optimization of both costs and performance [NTL05, JLB06, COG$^+$16], which is why new methods for assessing the status of the IM components are becoming vital in order to maximize availability and performance [S$^+$03, ZLHH09].

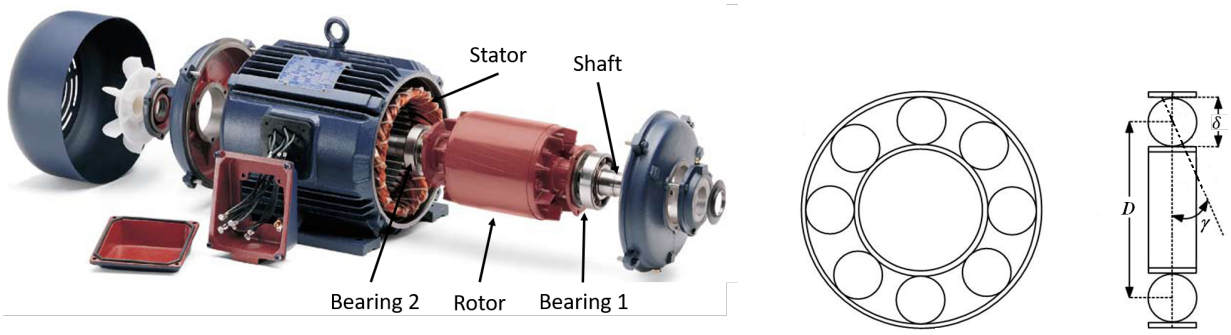
As far as IMs are concerned, the most vulnerable parts are the bearings, the stator winding, the rotor bar, and the shaft [KCMS16]. In particular, bearings play a primary role in the reliability and performance of an IM because they are subject to continuous mechanical stress and because they produce undesirable vibrations when decayed [ÖDS06]. Results on various studies (e.g. [SHKB95]) show that bearing decay account for the 41% of all IMs failures. Stator winding and rotor bar faults are responsible for respectively 37% and 10% of the total IMs faults. The remaining 12% of IMs faults are associated to other components (e.g. the shaft).

According to the literature on this subject [ICBR09, MD03, SABAA03], a common approach for monitoring the IMs bearing is to monitor the vibrations. By installing vibration sensors it is

possible to easily analyse fault signatures and salient fault features [GLLP04, PJH03, SMR07]. Unfortunately this procedure is not free from technical and economic downsides: placing sensors on the IM might not be effective nor economical. In fact, vibration sensor are not cheap, are prone to faults, are hard or impossible to install on many systems, and are sensible to corrosive and dusty environments [LCTH00, HXL+07, PCE+13]. An alternative way, with respect to study the vibration signals, is to study the stator currents [YKP+97]. This approach has many advantages respect to the previous one since it does not require the installation of any additional sensor and no direct access to the device is needed [Ben00, KS92]. In fact, the stator currents signals are already commonly available, or easily and unintrusively collectable [YKP+97].

Among the different techniques found in literature, developed to predict bearings damages from the stator current signal, two main approaches exists [NTL05, SSE03]. The first one is based on the analysis of the spectrum of the current and thrust while the second one is based on DA tools applied on the raw signals. In [BK03] it is provided a general review of the different frequency domain techniques applied to the analysis of motor currents and recently other approaches have been proposed [AOTR08, FB10, BGRR08, ZLHH09]. However, all these methods rely on a simple idea: collecting raw data, filter them and then apply frequency analysis in order to study the variations in the frequency spectrum before and after the damage is injected. The drawback of all these methods is the high dependence on the motor specific characteristics, whose parameters need to be known in advance, in order to determine a reference frequency spectrum function describing the bearings nominal state [SH97, ACS05]. Moreover, the frequency spectrum vibration analysis tend to be inaccurate if there are some slight variations on the load and current, due for example to different operative conditions or noise [SC07, ACS06]. In this context, many DA tools have been developed in order to overcome the limitation of the frequency-based approaches [BK03], since DA tools do not need any a-priory information about the IM and are more robust to noise [SLH+95]. Among the DA tools developed for bearings fault detection purposes different works have been proposed in literature [SLH+95, KOK03, FFTV00] but these methods do not exploit state-of-the-art approaches developed in the last years like the DNN. In fact, the inplicit feature learning conducted by DNNs, make them a good choice in this context [Ben09, BCV13, HOT06, HS06], since a more abstract representation of data is required.

In this work, a DNN is proposed for detecting the state of decay of the IM bearings able to extract from the stator current signal a compact and expressive representation of the bearings state, ultimately providing a bearing fault detection system. In order to estimate the effectiveness of the proposed procedure, a series of data from an inverter-fed motor with different damaged bearings was collected. As endurance tests would have resulted in the bearings break-down, thus compromising the approach test settings, it was decided to artificially induce different kinds of damage to the bearing in order to trace the evolution of their degradation state. The test was repeated with different motor loads for each damaged condition of the bearings, in order to asses the developed DNN in different operative conditions, according to the procedure described in described in Section 2.4.1.1. Results show that the proposed approach provides a promising and effective yet simple bearing fault detection system.



(a) (b)

Figure 3.9: (a) Induction Motor and (b) Bearing characteristic parameters

3.2.1 Induction Motors

Three-phase IMs are electrical machines characterised by low cost, ability to operate in hostile environments, good dynamic performance, and wide speed range operation. Such features make them a perfect candidate for industrial applications.

IMs are subjected to different modes of failures. The most common failure mode is the bearing failure, followed by stator winding and rotor bar failures. In Figure 3.9(a) an exploded view of an IM is reported. In Figure 3.9(b), the geometry of a Rolling-Element Bearing (REB) is reported. It consists of two rings (one inner and the other outer), where a set of rolling elements placed in the raceways rotates inside these rings [KS90]. Under normal operating conditions, considering a balanced load and a proper alignment, fatigue failure begins with small fissures, positioned below the rolling elements and slowly propagates to the external surface producing detectable vibrations and increasing noise levels.

From a mechanical point of view, local defects inside an IM bearing cause periodic impulses in vibration signals [KS90]. Period and amplitude of these impulses are in a relationship with the fault position, the shaft rotational speed and the bearing geometry [KS90]. REBs related defects can be categorized as outer bearing race defects, inner bearing race defects, ball defects, and train defects [Vas93]. The vibration frequencies to detect these faults can be described by the following relationships [SSÇU14]:

$$\begin{aligned} f_{od}&=\frac{n\cdot f_r}{2}\left(1-\frac{\delta}{D}\cos(\gamma)\right),\ f_{id}=\frac{n\cdot f_r}{2}\left(1+\frac{\delta}{D}\cos(\gamma)\right), \\ f_{bd}&=\frac{D\cdot f_r}{\delta}\left(1-\frac{\delta^2}{D^2}\cos^2(\gamma)\right),\ f_{tr}=\frac{f_r}{2}\left(1-\frac{\delta}{D}\cos(\gamma)\right), \end{aligned} \tag{3.16}$$

where f_{od} is the outer race defect frequency, f_{id} is the inner race defect frequency, f_{bd} is the ball defect frequency and f_{tr} is the train defect frequency. In these expressions, f_r is the shaft

rotation frequency, n the rollers number, δ and D are the roller and the pitch diameter of the bearing respectively and γ the contact angle, as depicted in Figure 3.9(b).

However, these equations require specific information concerning the bearing construction in order to calculate the exact characteristic frequencies. Nevertheless, for most bearings with $6 \leq n \leq 12$ [Sch90], the vibration frequencies can be approximated with:

$$f_{od} = \frac{2}{5} \cdot n \cdot f_r, \quad f_{id} = \frac{3}{5} \cdot n \cdot f_r. \tag{3.17}$$

In this way, it is possible to determine the bearing characteristic fault frequencies f_{od} and f_{id} without having explicit knowledge of the bearing construction.

Having this in mind, it is possible to retrieve these particular vibration frequencies in the spectrum, since a relationship between the bearing damage and the IM stator currents has been proved in [SHKB95]. In particular, the generation of rotating eccentricities at bearing fault characteristic frequencies f_c, leads to periodical changes in the machine inductance, producing additional frequencies f_{bf} in the stator current. Thus, any mechanical bearing vibration caused by a fault will produce a radial displacement between the rotor and stator. These air gap variations, caused by the bearing vibrations, affect the air gap permanence of the machine producing current harmonics at the following frequencies:

$$f_{bf} = |f_s \pm k f_c| \tag{3.18}$$

where f_s is the electrical stator supply frequency, f_c is the bearing fault characteristic frequencies, and $k = 1, 2, 3, \ldots$. The knowledge of the bearing characteristic frequencies and additional frequencies f_{bf} can be used to detect the cause of the defect, and examine single-point defects [SHH04].

Considering this information, it is possible to implement a monitoring system of the REB using vibration spectrum analysis to find the location, the cause, and the severity of defects. Traditionally, IMs condition monitoring was developed considering vibration and temperature measurements. The implementation of these systems could be expensive and is mainly adopted in the case of large motors or for critical applications [BGRR08]. Recently, most of the recent research [BGRR08, WCGT09] has been directed towards the monitoring of the IMs inspecting the phase current. In fact, the use of quantities that are already measured for command and control purposes in an IM, such as the machine's stator current, is favoured because allows the realization of cheaper, noninvasive and more reliable monitoring and diagnostic systems.

Moreover, since defects generate components in both vibration and current signals and the vibrations generated by the rolling-elect impact, have in the early stage relatively low energy, it is difficult to identify in the spectra using conventional frequency-based approaches, as it is proved in a preliminary frequency spectrum analysis in Section 3.2.2.

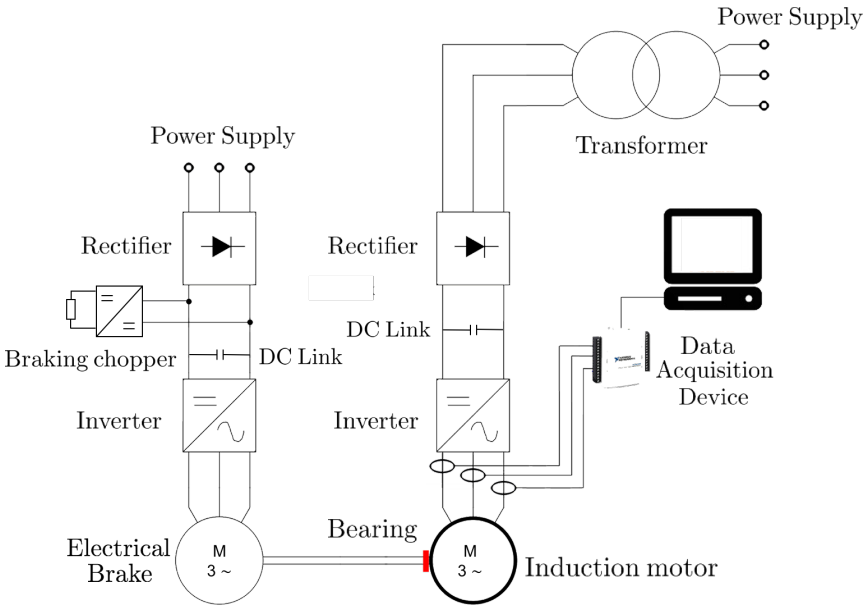


Figure 3.10: Principle scheme of the experimental setup.

Table 3.6: IM plate data.

	Value	Unit
Rated mechanical power P_n	1.5	[kW]
Rated line-to-line voltage V_n	230	[V]
Rated line current I_n	5.90	[A]
Polar pairs p	2	
Base frequency f_n	50	[Hz]

Table 3.7: Ball bearing parameters.

Bearing Parameter	Value
Outer diameter	52.0 mm
Pitch diameter D	38.5 mm
Roller diameter δ	7.9 mm
Rollers number n	8
Contact angle γ	0°

3.2.2 Experimental Campaign

To estimate the effectiveness of the proposed approach, an experimental campaign was conducted collecting the stator current signal from an inverter-fed IM mounting different artificially damaged bearings.

The bench set was designed within the University of Genoa, using a classic Motor-Transmission-User (MTU) system with a configuration that facilitates the bearings'replacement procedure. The experimental setup consists of a three-phase IM directly connected to a brushless motor, acting as an electrical brake. An inverter controls the driving IM. Figure 3.10 shows the experimental setup for collecting healthy and faulty stator current waveforms. In Table 3.6 the IM plate data are reported, while the parameters detail of the bearing used are in Table 3.7

The stator current signals have been collected from the current sensors connected to the power inverter. This configuration ensured the maximum distance between the current sensor and the

Figure 3.11: Bearings Condition: New (H0), $1.6mm$ hole (H1) and $5mm$ hole (H2).

motor, reducing the noise generated by the inverter magnetic field. Since it was not possible to carry out endurance tests, which would have led to the breakdown of the bearings, it was decided to introduce artificially damage into the bearing, to trace the evolution of the degradation state. The damage taken into account is located on the outer track of the bearing and can be easily inspected at the end of the tests to verify the actual condition.

Three identical bearings have been used covering the following bearing fault scenarios: no damages, size-1 artificial induced hole ($1.6mm$), size-2 artificial induced hole ($5mm$). These damages are lately respectively referenced as H0, H1, and H2. As a first step, the experiments were carried out for the healthy bearing condition to establish the base-line data. In Figure 3.11 the artificially damaged bearings condition is shown. For each bearing damaged condition, four different mechanical conditions have been investigated, applying to the motor shaft different resistive torques at the same rotational speed. In the following, each mechanical condition is identified by the corresponding stator current: 25%, 50%, 75% and 100% of the rated motor current I_n, and are lately referenced as L1, L2, L3 and L4. For each experiment, once the steady-state conditions have been reached, the stator currents have been acquired for 30 seconds. Experiments have been repeated many times for each damage condition, to build a large enough set of experiments.

The diagnostics of REB is generally performed by monitoring the vibrations of the bearing housing through accelerometers. The ISO 10816 standard [ISO16d] defines several classes, based on the overall vibration level, that can be used for an overall estimation of the system health status. The current-based bearing frequency spectrum monitoring could lead to discovering the presence of components in amplitude frequency correlated to the degradation state of the bearing as reported in Section 3.2.1. For single-point defects, the fault signatures can be discovered by monitoring their characteristic frequencies, although, they are subtle in the stator current signal where the dominant components are supply frequency components. In order to prove it, in this section, a preliminary frequency analysis of the stator currents data collected through the test campaign described in Section 3.2.1 is proposed. The characteristic frequencies f_{od}, f_{id}, f_{bd}, and f_{tr} computed using the formulas reported in paragraph 3.2.1, are reported in Table 3.8, while in Table 3.9 the frequencies $f_s \pm k f_{od}$ for k=1, 2, 3 are reported. In Figure 3.12, 3.13, and 3.14 the Fourier analysis of the stator current signal spectra of i_a are reported for the different defects H0,H1,H2, in the mechanical condition corresponding to the stator current equal to 100% of the

Table 3.8: Vibration Frequencies

Defect	Frequencies Hz
Outer race - f_{od}	79.5
Outer race (empirical) - f_{od}	80.1
Inner race - f_{id}	120.7
Inner race (empirical) - f_{id}	120.1
Ball defect - f_{bd}	103.5
Train defect - f_{tr}	8.8

Table 3.9: Characteristics Frequencies of Stator Currents

Defect	k = 1	k = 2	k = 3
Outer race $\|f_s \pm kf_{od}\|$	34.00 - 125.00	113.50 - 204.50	193.00 - 284.00
Inner race $\|f_s \pm kf_{id}\|$	75.24 - 166.25	196.00 - 289.00	316.70 - 407.70
Ball defect $\|f_s \pm kf_{bd}\|$	57.90 - 148.90	161.40 - 252.40	264.80 - 355.80

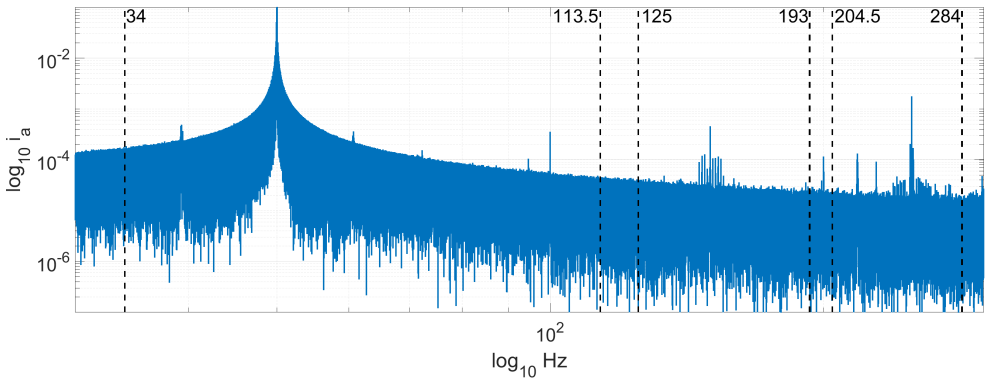
rated motor current I_n (L4). The same analysis can be repeated for the stator current i_b, producing results very similar to the ones obtained and reported for i_a, so, for brevity's sake, it was not reported here. From these figures, it is possible to observe that there are no visible variations between the spectra corresponding to states H0, H1, and H2, therefore it is not possible to find any correspondence between the calculated theoretical characteristic frequencies (reported with black dashed lines) and the frequencies obtained from the Fourier transform. The noise level present on the signal due to the use of the inverter does not allow to assess the degradation status by merely evaluating the amplitude of the frequency component. This result confirms the lack of effectiveness reached by state-of-the-art frequency analysis.

As a result, the use of a DNN is proposed for detecting the state of decay of the IM bearings able to extract from the stator current signal a compact and expressive representation of the bearings state, ultimately providing a bearing fault detection system.

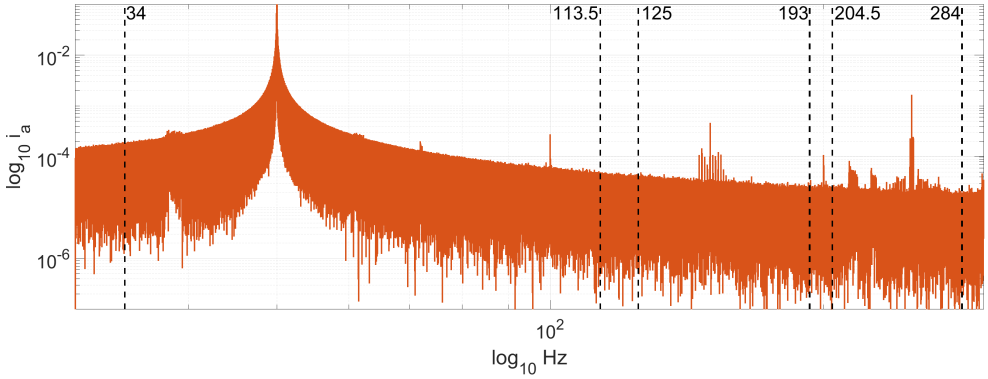
3.2.3 Data Analsysis Workflow and Results

In this section, the adopted workflow for the purpose of monitoring the bearings decay status together with the obtained results are reported. In particular the attention was focused on the following phases: (i) raw data collection and cleaning, (ii) data segmentation, (iii) initial simple FM, (iv) advance FM via deep unsupervised learning, and (v) bearings decay status prediction via deep supervised learning. These phases are depicted in Figure 3.15.

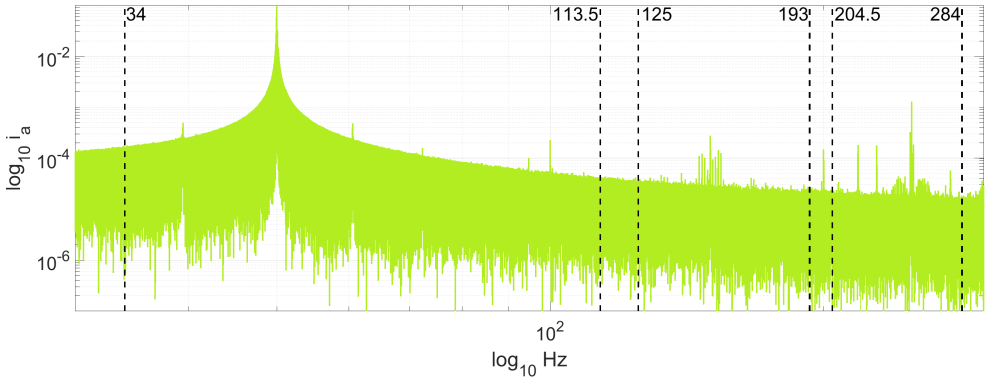
The first phase, described in details, consists in the process of collecting the raw data with an analog to digital converter device cleaned from the higher noise frequencies. The result of this process is a time series reporting, with a sampling frequency of $20KHz$, the value of two of



(a) H0, L4 - i_a - k=1

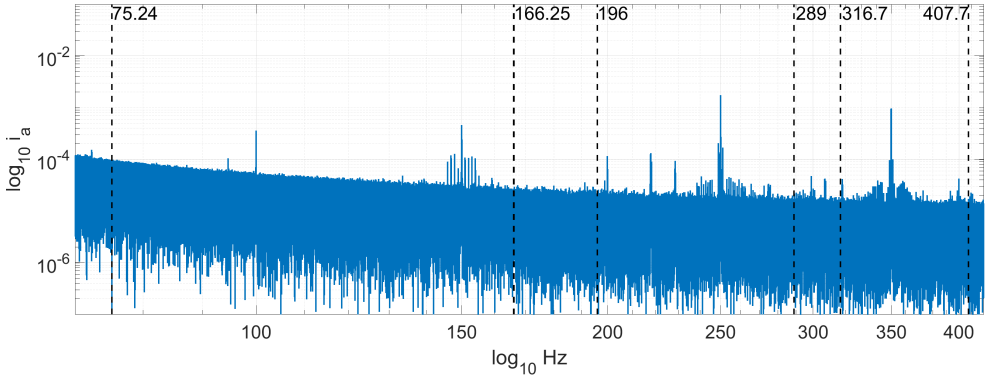


(b) H1, L4 - i_a - k=1

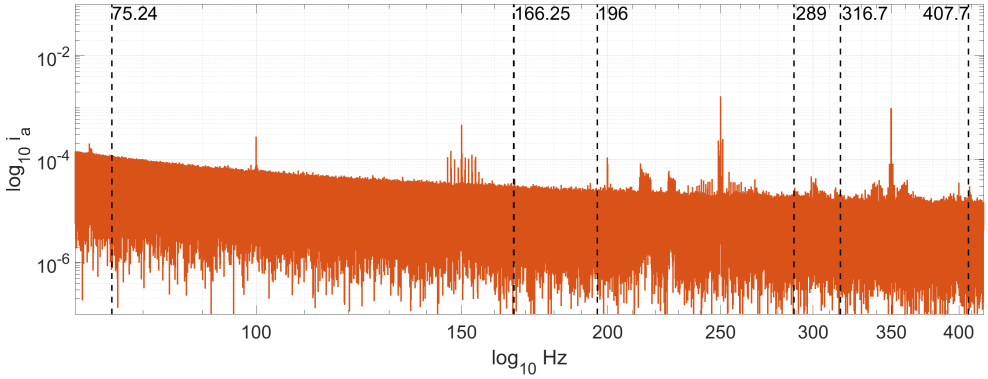


(c) H2, L4 - i_a - k=1

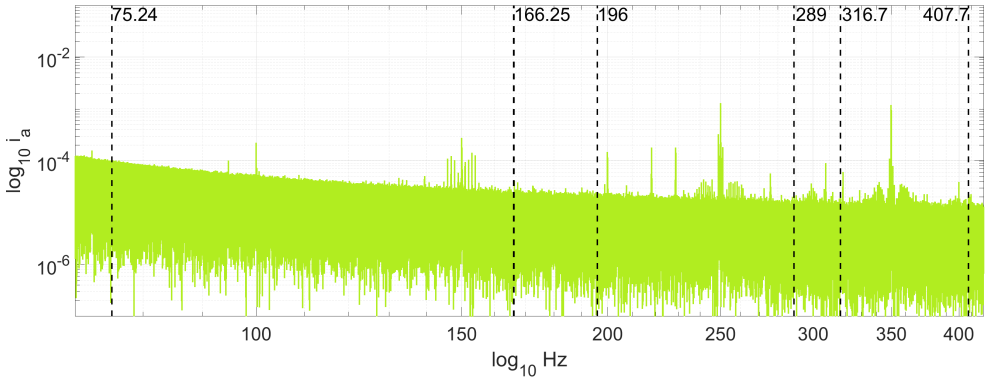
Figure 3.12: Frequency Analysis of i_a for the three different bearing defects H0, H1, and H2, considering L4 and $k = 1$.



(a) H0, L4 - i_a - k=2

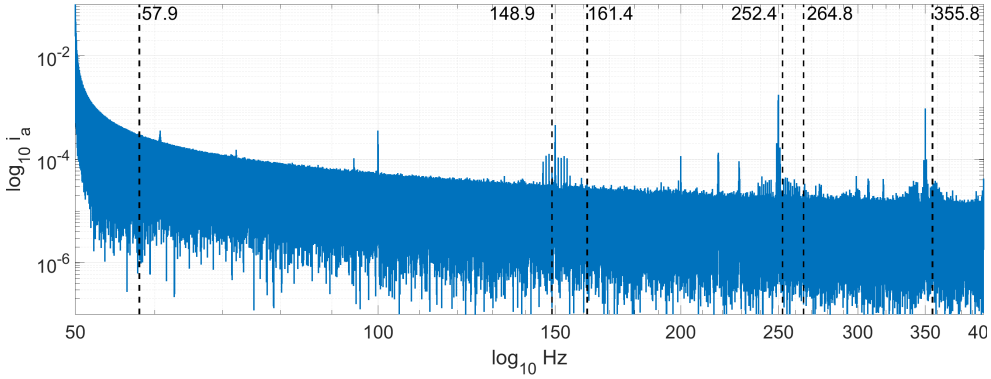


(b) H1, L4 - i_a - k=2

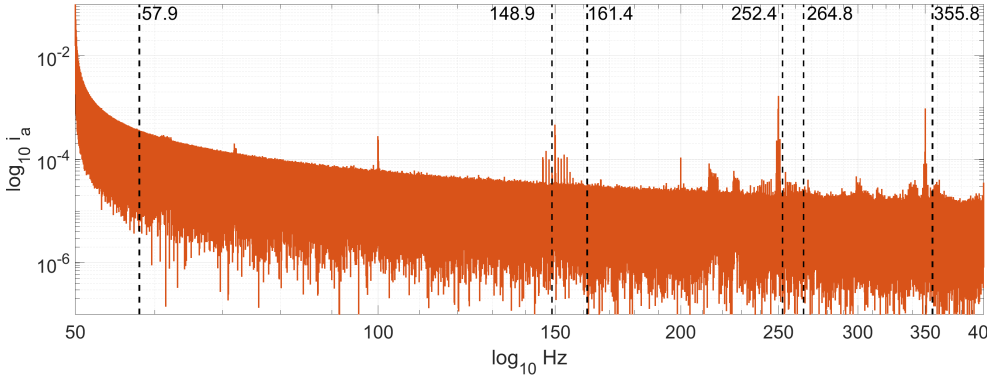


(c) H2, L4 - i_a - k=2

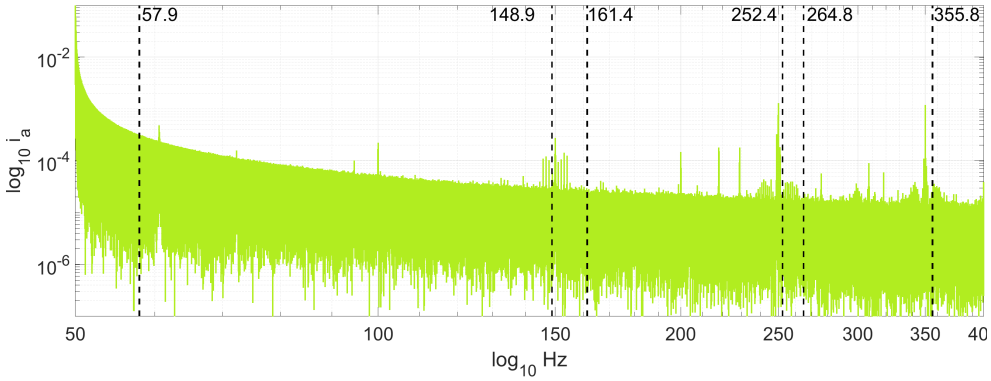
Figure 3.13: Frequency Analysis of i_a for the three different bearing defects H0, H1, and H2, considering L4 and $k = 2$.



(a) H0, L4 - i_a - k=3



(b) H1, L4 - i_a - k=3



(c) H2, L4 - i_a - k=3

Figure 3.14: Frequency Analysis of i_a for the three different bearing defects H0, H1, and H2, considering L4 and $k = 3$.

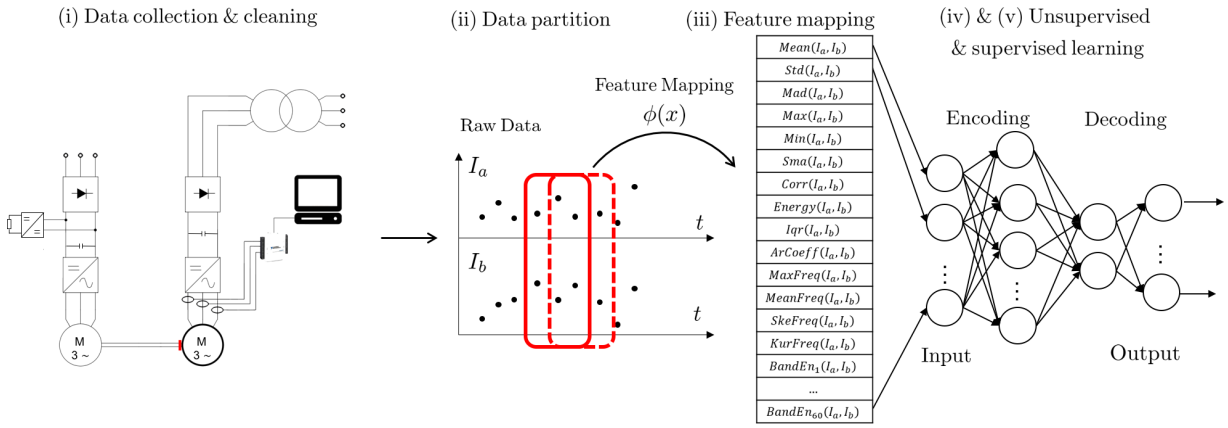


Figure 3.15: Bearings decay status monitoring system workflow.

Table 3.10: Simple feature set extracted from the windowed raw data.

Time Domain	Frequency Domain
Signal magnitude area of i_a and i_b	Largest frequency component of i_a and i_b
Correlation coefficient between i_a and i_b	Frequency signal average of i_a and i_b
Average sum of the squares of i_a and i_b	Frequency signal Skeewness of i_a and i_b
Interquartile range of i_a and i_b	Frequency signal Kurtosis of i_a and i_b
Signal Entropy of i_a and i_b	Energy at 60 band frequencies of i_a and i_b
Autoregression coefficients of i_a and i_b	
Mean value of i_a and i_b	
Standard deviation of i_a and i_b	
Median absolute value of i_a and i_b	
Largest values of i_a and i_b	
Smallest value of i_a and i_b	

the three stator currents, i_a and i_b, stated that just two of them, i_a and i_b, have been utilised to develop the proposed ML approach.

The second phase consists in segmenting these raw data in overlapping sliding time windows of $24s$. This quantity has been selected considering the peculiar characteristic of the studied IM, so to have a window large enough to capture the dynamics of the IM.

In the third phase, a series of simple yet informative features was extracted from the windowed raw data, which have been chosen based on previous studies on similar context [AGO+13]. The list of these features is reported in Table 3.10. The result of this FM is a sample $\boldsymbol{x} \in \mathcal{X} \subseteq \mathbb{R}^d$ with $d = 155$ (see Table 3.10) with associated its label $\boldsymbol{y} \in \mathcal{Y}$, where $y_1 \in \mathcal{Y}_1 = \{1, 2, 3\}$ represents the decay state of the bearing (see Figure 3.11) and $y_2 \in \mathcal{Y}_2 \subseteq \mathbb{R}$ represents the load level. For each experiment and each window, a different sample which composes the data $\mathcal{D}_n = \{(\boldsymbol{x}_1, \boldsymbol{y}_1), \cdots, (\boldsymbol{x}_n, \boldsymbol{y}_n)\}$ is then provided. Based on the experiments described a total of $n = 1400$ samples have been collected. Consequently, this problem can be mapped into a

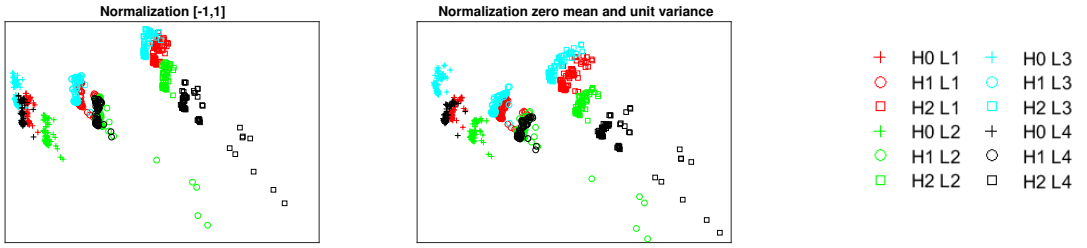


Figure 3.16: Projected test point in the two-dimensional space defined by the two most informative PCA.

multioutput (two labels) and multitasks problem (one label brings to a classification task while the other to a regression one) [STC04].

Unfortunately, even if the simple FM of Table 3.10 is quite informative, it is characterized by some drawbacks. The first one is that this FM is quite high dimensional and consequently hard to interpret for a human operator. The second one is that due to this high dimensional space and the low number of experiments, and consequently low number of samples, the risk is to overfit the available data instead of learning some meaningful information out of them.

To overcome these issues, an unsupervised dimensionality reduction approach must be applied to reduce the space and do not overfit the data. A simple approach is to use the Principal Component Analysis (PCA) [Pea01]. PCA assumes that data lie in a low dimensional informative space, which have been roto-translated in a higher dimensional space. PCA can be thought of as fitting an n-dimensional ellipsoid over the data. Each axis of the ellipsoid represents a new component. The larger is the axis, the higher is the variance of the data along that dimension and, consequently, the more relevant is that component as it varies more. In other words, the components with low variance are less informative. Unfortunately, this approach is too naive, and also scaling-dependent. Hence, PCA did not allow to obtain an informative low dimensional representation of the data.

For this reason, a SNN and a DNN autoencoder were exploited as an unsupervised dimensionality reduction approach as described in Section 2.4.1 and shown in Figure 2.1.c.

In the following, the results obtained by adopting the DA workflow are reported, over the data collected during the trial, to obtain an IM bearings fault detection system.

The main purpose of this study is to obtain, from the initial simple FM, a compact and expressive representation of the bearings state. For this purpose, the first approach the PCA was adopted for reducing the dimensionality of the original data to check whether, in a lower dimensional space that can be interpreted by an operator, it is possible to give a meaningful low dimensional representation of the IMs bearing fault phenomena. Since PCA is scaling-sensitive, many normalization methods were tested. Figure 3.16 reports the results obtained via PCA with different normalization methods. From Figure 3.16 it possible to note that PCA it is not able to repre-

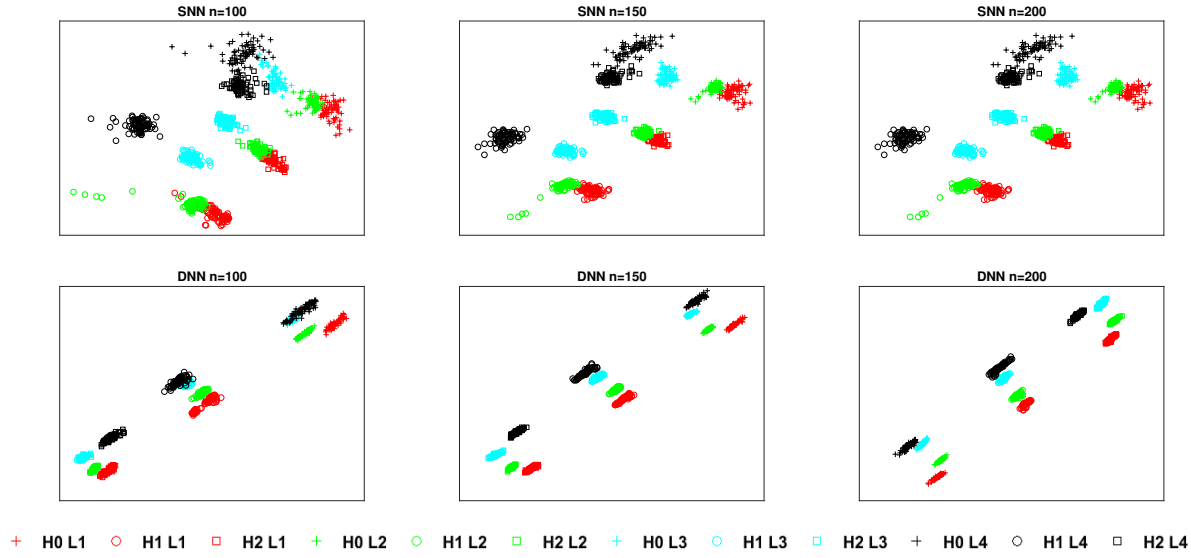


Figure 3.17: Projected test point in the two-dimensional space defined by the different networks.

sent the phenomena in a two-dimensional space since the data cloud of the different operational conditions (faults H0, H1, and H2 and loads L1, L2, L3, and L4) are overlapped.

Based on the result obtained with the PCA, an SNN and then a DNN were exploited trying to compress all the information in the two neurons in the second-last layer. Moreover, to simulate in a more realistic scenario, it was assumed to have a reduced amount of labeled samples. For this reason just $n \in \{100, 150, 200\}$ labeled samples were exploited to train the network and in Figure 3.17 the projected test points in the two-dimensional space, defined be the different networks, is depicted. Note that the training and the validation phases have been performed. From Figure 3.17 it is possible to observe that:

- DNN and partially SNN, differently from PCA, are able to find a compact and expressive representations of the bearings damage status, by grouping the data in separate clusters based on load and damage conditions;
- both in SNN and DNN learned representation groups are ordered by load and entity of the damage;
- DNN provides clearer and more defined clusters with respect to SNN, showing higher classification performances even when the number of training samples is extremely limited.

As a consequence, based on the reported results, it is possible to state that the DNNs are able to extract from the stator current signal a compact and expressive representation of the bearings state, ultimately providing a bearing fault detection system.

Table 3.11: Influence of Hull [Sch07].

Visual Hull Condition	% Power Increase
Freshly applied hull coating	0
Deteriorated coating or slime	9
Heavy slime	19
Small calcareous fouling or macroalgae	33
Medium calcareous fouling	84

3.3 Data-Driven Ship Digital Twin for Estimating the Speed Loss due to the Marine Fouling.

Sustainable shipping is recognised as one of the biggest challenges of the 21st century[2]. Approximately 90% of world trade is transported on water and even if shipping is relatively safe and clean, compared with other means of transportation, it has a significant impact on the environment. The precautionary principle, sustainable development policies and ideals, greater public concern about global environmental issues, and pressure from other sectors, all serve to reinforce the need for the shipping industry to behave in a more sustainable manner [Cra93, WSF15, Nik13, SJA+14]. As a consequence, developing new technologies able to both improve the design of the ships and to maintain their efficiency becomes a crucial issue.

In this work, the attention was focused on the problem of keeping the ship as much efficient as possible by estimating the degradation state of its components, with the consequent performance loss and fuel consumption increase. Broadly speaking, as far as propulsion systems are concerned, there are mainly three macro-components in a ship that can degrade: the main engine, the hull, and the propeller [Cal92, Car12, ACJW18]. Apart from the ordinary regular maintenance, the main engine degrades very slowly in time and related effects are only noticeable after years of operations [Cal92]. The hull and the propeller, instead, are subject to marine fouling, that increases the frictional resistance of the parts moving through the water and, hence, decreases their efficiency. The effects of marine fouling can be clearly observed after just a few months of operations [PK14, DUZ+17]. Marine fouling, or simply bifouling, is defined as the undesirable accumulation of microorganisms, algae, and animals on artificial surfaces immersed in seawater [YKDJ04, ABLWB16]. On the hull, the presence of fouling increases the roughness of the surface, hence increasing frictional resistance. Hull resistance accounts for the 90% of the total ship resistance [Sch04, Kem37] and the fouling can nearly double it (see Table 3.11 taken from [Sch07]). On the propeller, the presence of fouling increases the roughness of the blade surface, thus requiring more power to maintain the same speed [AGC+02, SAG16, ODO+18]. Fouling represents the primary cause of hull [CAA03] and propeller [KX11] performance degradation.

[2] http://www.ssi2040.org

Currently, shipping companies try to mitigate the problem of hull and propeller fouling by applying anti-fouling paints on the submerged surfaces and by regularly cleaning the hull. Despite their effectiveness, such methods have some drawbacks. In spite of their prime role and effectiveness in preventing fouling growth, depending on their types, antifouling paints can be expensive (e.g. non-biocidal Fouling Release type) and can be harmful to the marine environment (e.g. biocidal Self-Polishing types). Moreover, the hull and the propeller are cleaned on the occasion of other dry-docking maintenance events, but this practice does not ensure an optimal scheduling of the cleaning procedures. A reliable and effective planning of these activities should take into account the speed loss caused by the fouling, to find the optimal balance between efficiency and costs. For this reason an accurate estimation of the speed loss caused by fouling is needed [Sch07, AYT+18, ODO+18].

However, providing a quantitative estimation of the speed loss associated to the fouling phenomenon is a challenging task [DUZ+17, DTI17, CPA17]. The latter depends on many factors, such as the speed and the draft of the ship, the sea state, the wind speed and direction, etc. Furthermore, the accumulation of marine organisms on the hull is faster when a vessel is frequently in harbour, stationary, or in high-temperature tropical waters [Ste37].

The state-of-the-art approach for estimating the speed loss can be carried out by applying the standard ISO 19030 [ISO16b] proposed by the International Organization for Standardization (ISO). The ISO 19030 prescribes methods for measuring changes in hull and propeller performance and it defines a set of relevant performance indicators for their maintenance, repair, and retrofit activities. Specifically, the ISO 19030 suggests comparing the measured performances with the ones obtained during sea trials in particular operating points. This comparison provides an indicator of the hull and propeller efficiency. A continuous monitoring of the efficiency provides a reliable estimation of the changes in the performances.

Despite its simplicity and effectiveness, the ISO 19030 presents some limitations. The procedure requires filtering out operating points that are outside the prescribed boundaries, thus limiting the ability of the method to monitor the ship over a wide set of operating conditions [KBK18]. Moreover, some corrections are needed to cope with the environmental disturbances (i.e. winds, waves, and currents). Unfortunately, these corrections require the use of complex fluid dynamics models or additional sea trials [ISO16a, ISO16b, ISO16c].

Some attempts have been made to address the ISO 19030 limitations. [Log12] uses measurements of the propeller performance as efficiency indicators; however, this procedure requires the exclusion of many operating points to eliminate the effects of current, ship motions, rudder, and transients, with techniques similar to the ones reported in the ISO 19030 and with all their inconveniences. [BK16] propose an operational approach for obtaining an accurate fuel consumption and speed curve, parametrised for the major influence factors, such as ship's draft and displacement, waves forces and directions, hull and propeller roughness. They propose a statistical analysis on 418 noon reports for a case-study vessel to consider the influence of the above factors in fuel consumption. The proposed approach, similarly to the ISO 19030 procedure, re-

lies on simplified corrections for environmental disturbances, draught, and speed. This applies also for the work proposed by [FTK17], whose method is based on a correction of measured data based on a PM of the influence of wind and waves on ship performance. While models based on the physical knowledge of the problem are well-established [Log12, ISO16a], they often fail in predicting the effect of ship-specific and environmental phenomena.

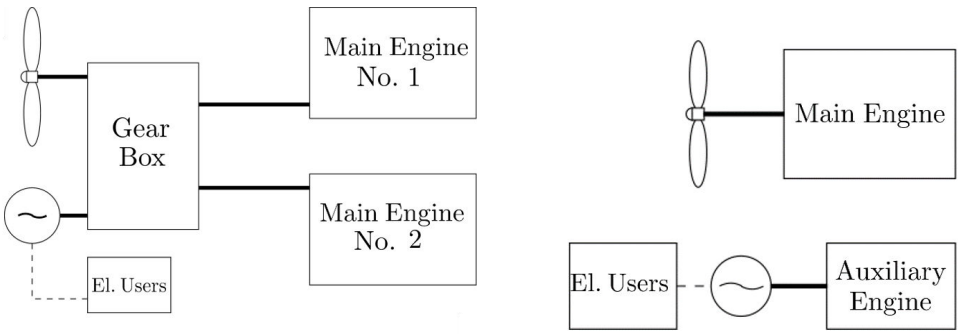
On the other hand, DDMs can easily take into account many phenomena thanks to the use of large amount of data with little, if no physical knowledge about the problem. DDMs have shown to be an effective tool for the solution of many problems in the shipping industry, and, in particular, NNs and Gaussian Process were employed to estimate the ship's fuel consumption efficiency in [PL09] and [PJW12], while in [RV15] a NN Ensemble is exploited for towboat shaft power prediction. In [COBA17] it is shown how DDMs outperform PMs in the prediction of ship performance. Finally, in [LSSV08] the performances of white, grey, and black box models for predicting the fuel consumption are tested.

Deep Learning techniques represent the state-of-the-art for dealing with data driven problems, but require a fine-tuning procedure to set all the hidden parameters in the architecture and are affected by the problem of local minima and slow convergence rate: this results in a cumbersome and time-consuming procedure [KZHV13, TDH16]. In order to overcome these limitations, a successful proposal is the one of [TDH16] which develops a deep architecture which exploits the principle behind the DELM reported in Section 2.4.1.2.

In this work, the use of DELM is porposed for estimating the speed loss caused by the marine fouling effects on the ship hull and propeller, leveraging on the large amount of information collected from the on-board monitoring system sensors. Inspired by the ISO 19030 and supported by the evidence that DDMs can be much more accurate and effective than the physical ones, a DDM was proposed for predicting the speed of the ship, able to act as a "Digital Twin" [GS12, BR16] of the ship itself. The Digital Twin can be used to compute the deviation between the predicted performance and the actual one, namely the speed loss. It was shown that the average drift in time of the speed loss can be exploited to accurately and effectively estimate the effects of the marine fouling on the ship performance. To this aim, they propose a two-phase approach:

(I) firstly, a DDM based Digital Twin is built, leveraging on the large amount of information collected from the on-board monitoring system sensors;
(II) secondly, the same model is applied in order to estimate the speed-loss of the ship and its drift.

Obviously the Digital Twin needs to be tuned on data collected during a period of time where the marine fouling is not present and for a time period wide enough to observe the ship in many operational and environmental conditions: data collection can start just after the launch of the ship (or after hull and propeller cleaning) and stop after some months of operations. Deep Learning techniques represent the state-of-the-art for dealing with Phase (I). For what concerns Phase (II), instead, it is show that the average behaviour of the speed loss between two maintenance events



(a) V1 (b) V2

Figure 3.18: Conceptual representation of the propulsion systems of the V1 and the V2.

(where also hull and propeller cleaning is performed) is characterized by a clear drift, easily detectable with a robust regression [ZS10] in time of the predicted speed losses. Moreover, it is shown, by means of the nonparametric statistical test of Kolmogorov-Smirnov [Smi44], that the distribution of the speed loss before and after two maintenance events changes in a statistical significant way while, during the operations, such a distribution changes smoothly.

A comparison between the proposed method and the ISO 19030 on real-world data coming from two Handymax chemical/product tankers has been carried out and is presented in this work to show the effectiveness of the proposal.

3.3.1 Available Dataset

In this section the data coming from the two Handymax chemical/product tankers data logging systems are reported. The first vessel (V1) was designed and built for transporting chemicals and petroleum products up to $46764DWT$ with a design speed of $15knots$. The vessel is $176.75m$ long (between perpendicular) and $32.18m$ wide, run by two four-stroke engines providing a total propulsive power of $7680kW$. The second vessel (V2) is a tanker for chemicals and oil products up to $46067DWT$ with a design speed of $15.5knots$. The vessel is $176.83m$ long (between perpendicular) and $32.20m$ wide, run by a two-stroke engine power of $8200kW$. A conceptual representation of the propulsion system of the two vessels is shown in Figure 3.18, while their main features are presented in Table 3.12.

The V1 is equipped with two main engines (MaK 8M32C four-stroke Diesel engines rated $3840kW$) and designed for operation at $600rpm$. The engines are connected to a gearbox that distributes the power between the controllable pitch propeller for propulsion and a shaft generator (rated $3200kW$). Auxiliary power can also be generated by two auxiliary engines rated $682kW$ each. Each main engine is equipped with exhaust gas boiler, that can be integrated with two auxiliary oil-fired boilers.

Table 3.12: Main features of the V1 and the V2 case studies.

	V1		V2	
Ship Feature	Value	Unit	Value	Unit
Deadweight	46764	$[t]$	46067	$[t]$
Design speed	15	$[knots]$	15.5	$[knots]$
Draft (summer SW)	12.18	$[m]$	12.2	$[m]$
Length between perpendicular	176.75	$[m]$	176.83	$[m]$
Breadth moulded	32.18	$[m]$	32.20	$[m]$
Main engines installed power	3840×2	$[kW]$	8200	$[kW]$
Auxiliary engines installed power	682×2	$[kW]$	1176×3	$[kW]$
Shaft generator power	3200	$[kW]$		
Exhaust boilers steam generator	750×2	$[kg/h]$	1130	$[kg/h]$
Auxiliary boilers steam generator	14000×2	$[kg/h]$	14000×2	$[kg/h]$
Fuel consumption	34.7	$[mt/day]$	31.8	$[mt/day]$

The V2 is equipped with one main engine (MAN B&W 6S50MC slow speed, two-stroke engine rated $8200kW$) and designed for operation at $120rpm$. In this case, the auxiliary power is generated by three Diesel-generators rated $1176kW$ each. As for V1, the main engine is equipped with an exhaust gas boiler, that can be integrated with two auxiliary oil-fired boilers.

The two vessels are equipped with the same data logging system which is used by the company for both on board monitoring and land-based performance control. Table 3.13 summarizes the available measurements from the continuous monitoring system. The original frequency of data acquisition by the monitoring system is equal to 1 point every 15 seconds. In order to provide easier data handling, the raw data are sent to the provider server, where they are processed to collect a set of 15 minutes averages. In this work, the latter dataset was used for the application of the proposed method.

The available data of the two vessels have been collected during the time slots reported in Table 3.14. Note that the data are characterized by many missing points due to failure in the data logging system, or maintenance, or stops of the vessels.

3.3.2 The ISO 19030

In order to check the effectiveness of the proposed method against a state-of-the-art approach, the procedure suggested by the ISO 19030 was implemented for monitoring hull and propeller performance [ISO16a, ISO16b, ISO16c]. In this section, the application procedure proposed by the ISO 19030 is presented from an operating point of view. More details are available in the reference documents [ISO15]. The application of the ISO 19030 procedure, given the information collected from the data logging system as reported in Table 3.13, can be summarized in the following steps:

Table 3.13: Data collected from logging system of the two vessels.

Variable name	Unit	Variable name	Unit
Timestamp	$[t]$	Sea depth	$[m]$
Latitude	$[^\circ]$	Seawater temperature	$[^\circ C]$
Longitude	$[^\circ]$	CPP set point	$[^\circ]$
Main engines fuel consumption	$[kg/h]$	CPP feedback	$[^\circ]$
Auxiliary engines power output	$[kg/h]$	Fuel density	$[kg/m^3]$
Shaft generator power	$[kg/h]$	Fuel temperature	$[^\circ C]$
Propeller shaft power	$[kW]$	Ambient pressure	$[bar]$
Propeller speed	$[rpm]$	Humidity	[]
Ship draft (fore)	$[m]$	Dew point temperature	[]
Ship draft (aft)	$[m]$	Shaft torque	$[kNm]$
Draft port	$[m]$	Rudder angle	$[^\circ]$
Draft starboard	$[m]$	Acceleration x direction	$[m/s^2]$
Relative wind speed	$[m/s]$	Acceleration y direction	$[m/s^2]$
Relative wind direction	$[^\circ]$	Acceleration z direction	$[m/s^2]$
GPS heading	$[^\circ]$	Roll	$[^\circ]$
Speed over ground	$[knots]$	Pitch	$[^\circ]$
Speed through water	$[knots]$	Yaw	$[^\circ]$

Table 3.14: Data collection time interval for the two vessels.

Vessel	Data Collection Start	Data Collection End
V1	21/03/2012 17:45:00	03/10/2014 14:15:00
V2	01/05/2014 00:15:00	26/08/2016 14:15:00

(I) Data filtering
(II) Correction for environmental factors
(III) Calculation of Performance Values (PVs)
(IV) Calculation of Performance Indicators (PIs)

Step (I) is performed by applying the Chauvenet's criterion [Cha63] to all measured variables, according to which a datum is to be considered an outlier if:

$$\text{erfc}\left(\frac{\Delta_i}{\sigma\sqrt{2}}\right) N < 0.5 \tag{3.19}$$

where erfc is the complementary error function [Gla71], Δ_i represents the difference between the i-th datum and the mean value over the dataset, σ is the standard deviation of the variable of interest, and N the size of the dataset. In addition, further filtering was applied considering outliers also points for which:

$$v_v < 8[knots], \quad |v_w| > 8[m/s] \tag{3.20}$$

where v_v and v_w are the speed of vessel and wind respectively. The additional filtering on the ship speed was added in order to avoid numerical errors in the evaluation of the speed loss for low values in the denominator of Eq. (3.24), while the filter on the wind speed was added to filter out points with bad weather conditions, since the behaviour of the vessel in those conditions is strongly inconstant and unreliable. Step (II) included the power correction Δ_P based on measurements of wind speed and direction:

$$\Delta_P = (R_w - R_{0w}) v_v^2 \frac{1}{\eta_{0p}} + P_p(1 - \frac{\eta_p}{\eta_{op}}) \tag{3.21}$$

where R_w represents the ship's wind resistance due to relative wind, R_{0w} is the air resistance in no-wind conditions, P_p is the propulsive power, η_p its propulsive efficiency, and η_{0p} is the propulsive efficiency in calm condition. In absence of more accurate information, η_p is set to 0.7, as suggested by the ISO. The ship wind resistance R_{rw} is computed as follows:

$$R_{rw} = 0.5\rho_a v_w^2 A_{tp} C_w(\psi_w) \tag{3.22}$$

where ρ_a is the air density, A_{tp} is the transverse projected area, C_w is the wind resistance coefficient, and ψ_w is the wind relative direction. Eq. (3.22) is used for calculating both the actual and the reference wind resistance using the relative wind speed and the relative wind direction in the first case and the ship speed and head wind direction in the second case. The wind resistance coefficient is computed based on [FUI06].

Step (III) involves the calculation of the percentage speed loss based on the corrected propulsion power. The expected speed v_{exp} is computed based on reference, clean-hull data interpolated starting from actual measurements of draft (T) and trim (δ):

$$v_{exp} = f(P'_p, T, \delta) \tag{3.23}$$

Table 3.15: Maintenance events for the V1 and the V2.

V1

Date	Event
21/03/2012	Vessel delivery
29/10/2012	Propeller cleaning
30/03/2013	Hull cleaning
01/08/2013	Loss of the LOG speed measurement
17/07/2014	Change from fixed-speed to variable-speed operations

V2

Date	Event
19/04/2014	Propeller polishing
20/12/2014	Hull cleaning
28/08/2015	Hull cleaning and Propeller polishing
28/11/2015	Dry-docking

where P'_p is the corrected power for accounting the effect of the draft and trim. This allows to compute the percentage speed loss $\mathrm{SL}_{\%}$ as:

$$\mathrm{SL}_{\%} = 100\frac{v_m - v_{exp}}{v_{exp}} \tag{3.24}$$

where v_m is the measured speed.

The speed loss is then used as performance value for the calculation of the different performance indicator in Step (IV). The ISO procedure suggests comparing the average value of the speed loss over a given period of time in order to average out uncertainties and statistically not-relevant fluctuations.

3.3.3 Results Comparison

In this section, the comparison of the DELM and ISO 19030 results for the estimation of the speed loss due to fouling is reported.

Firstly, an analysis of the drift in DELM and ISO 19030 estimated percentage speed loss between two consecutive hull and propeller cleaning events, carried out with the linear robust regression is presented. Secondly, the analysis of the changes in time of DELM and ISO 19030 estimated percentage speed loss distributions, carried out with the nonparametric statistical test of Kolmogorov-Smirnov. Table 3.15 reports the recorded relevant maintenance events of the two vessels.

Figures 3.19 and 3.20 report the results drift in DELM and ISO 19030 estimated percentage speed loss between two consecutive hull and propeller cleaning events for the V1 and the V2

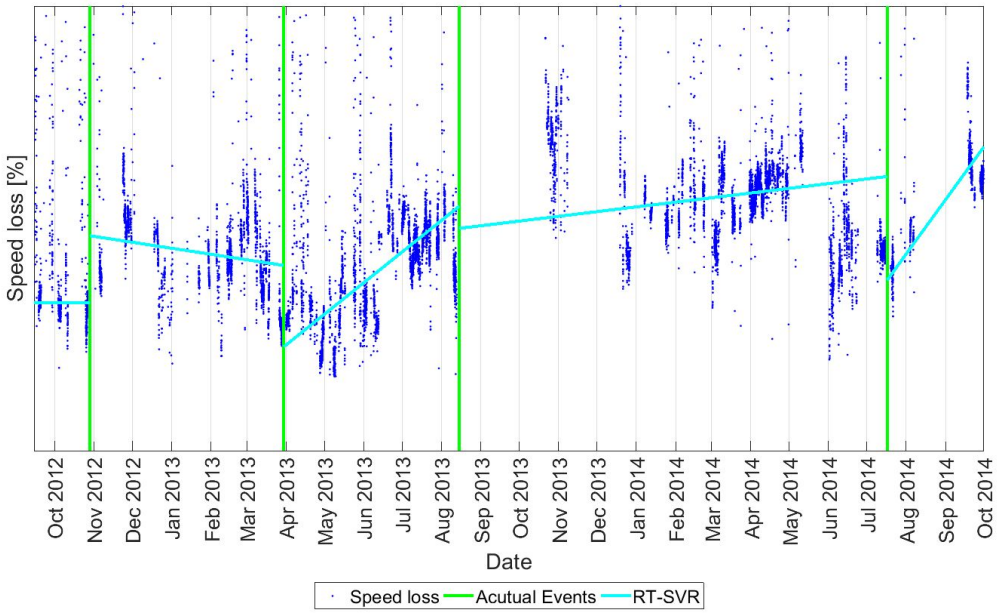
respectively. Those results clearly show the higher level of reliability of the prediction achieved by the DELM method against the ISO 19030 one.

In both vessels, the linear trend for the speed loss calculated by the ISO 19030 method shows large variations between different maintenance intervals (particularly for the V1, but also for the V2). In addition, in some intervals between two consecutive hull and propeller cleaning operations, the trend in the estimated percentage speed loss using the ISO 19030 method is negative. These results do not agree with the physical basis of the fouling phenomenon, and suggest that, in the case presented in this work, the application of the ISO can lead to inaccurate results.

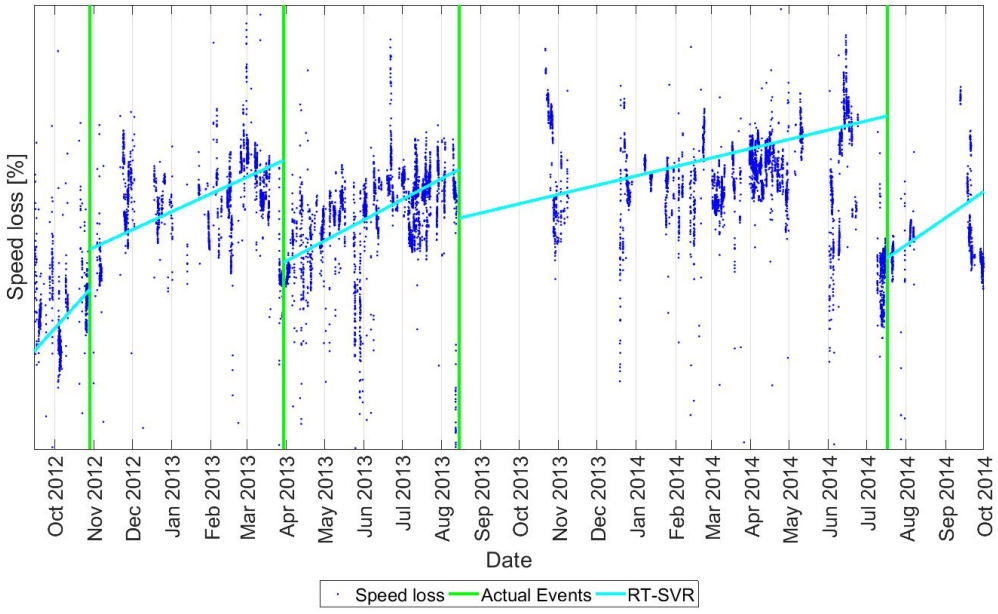
On the contrary, as far as DELM is concerned, Figures 3.19 and 3.20 clearly show trends that are always physically plausible. Model drift behavior between different cleaning intervals is now consistent with the one characterizing a ship that operates in conditions, on average, similar over time.

Figures 3.21 and 3.22 report the changes in time of DELM and ISO 19030 estimated percentage speed loss distributions of the V1 and the V2 respectively. Those figures tesfify the higher level of reliability of the DELM method against the ISO 19030 one. In both vessels there is no statistically meaningful changes in the distribution of the speed losses estimated with the ISO 19030, and just in a few cases the Kolmogorov-Smirnov test detects a change in correspondence to an actual hull and propeller cleaning event (see Table 3.15). On the contrary, when the same method is applied to the speed losses estimated with the DELM approach, the Kolmogorov-Smirnov test detects all the changes in correspondence or close to an actual hull and propeller cleaning event. From the obtained results, it is possible to firmly conclude that the proposed method based on DELM can provide a more accurate estimation of the changes in performance due to hull and propeller marine fouling with respect to the ISO 19030 based model.

As showed previously, this method represents an improvement compared to ISO standards from many points of views, while only requiring operational data as input (hence no model tests, ship design data, etc.). The need for an efficient calculation method for the effect of fouling on ship performance has been highlighted by many actors in shipping, both for economical and for environmental reasons [CSC11]. The speed loss calculated using the linear robust regression as showed in Figures 3.19 and 3.20 provides an accurate picture of the status of the hull and propeller fouling at a specific point in time. This information could be used effectively to optimize the scheduling of maintenance events. Today, hull and propeller cleaning are performed at fixed intervals, or in correspondence of other maintenance events (e.g. dry-docking). In practice, they could be performed more or less often depending on the actual status of the hull and propeller, according to methods based on the minimization of costs, fuel consumption, and emissions. Based on a similar principle, the method could be used as a mean for the critical evaluation of interventions directed towards the limitation of fouling effect. Measuring the value of the speed loss before, and after, maintenance events can help to understand their effectiveness, and hence making more appropriate and informed choices. As an example, the hull cleaning performed on

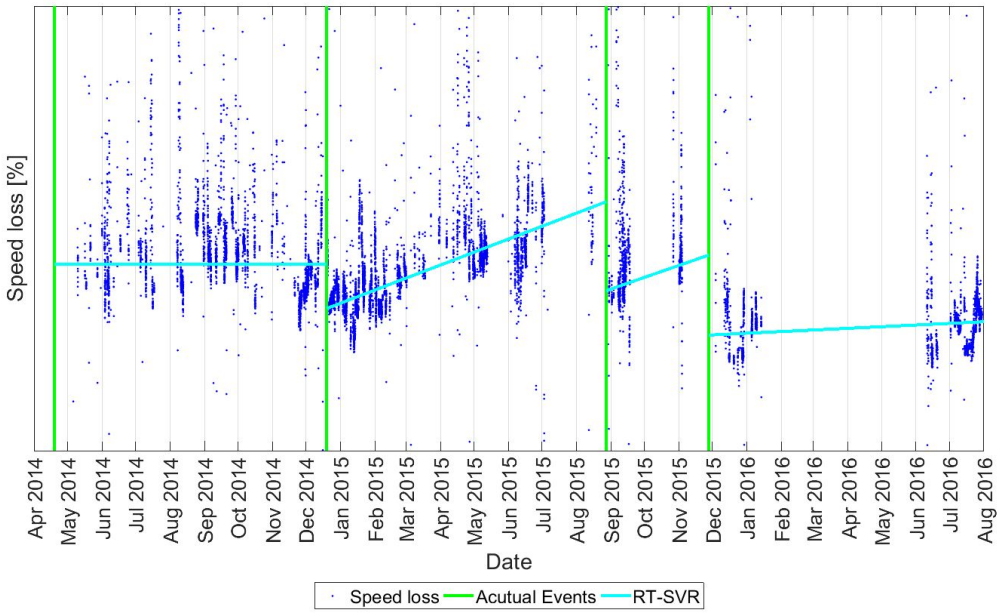


(a) ISO 19030

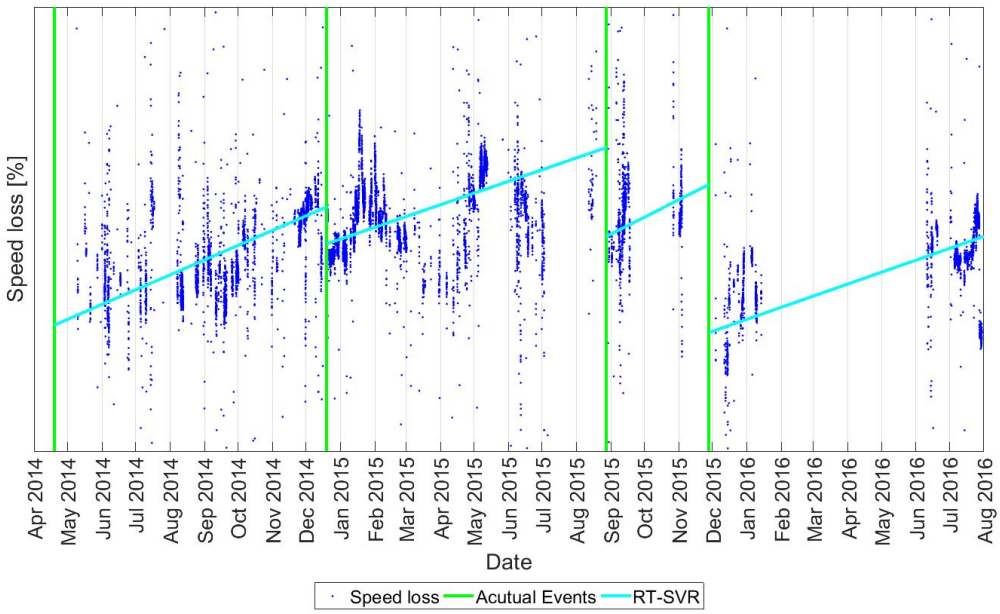


(b) DELM

Figure 3.19: Linear Robust Regression on the Speed Loss Percentages between two consecutive Hull and Propeller Cleaning Events for the V1.

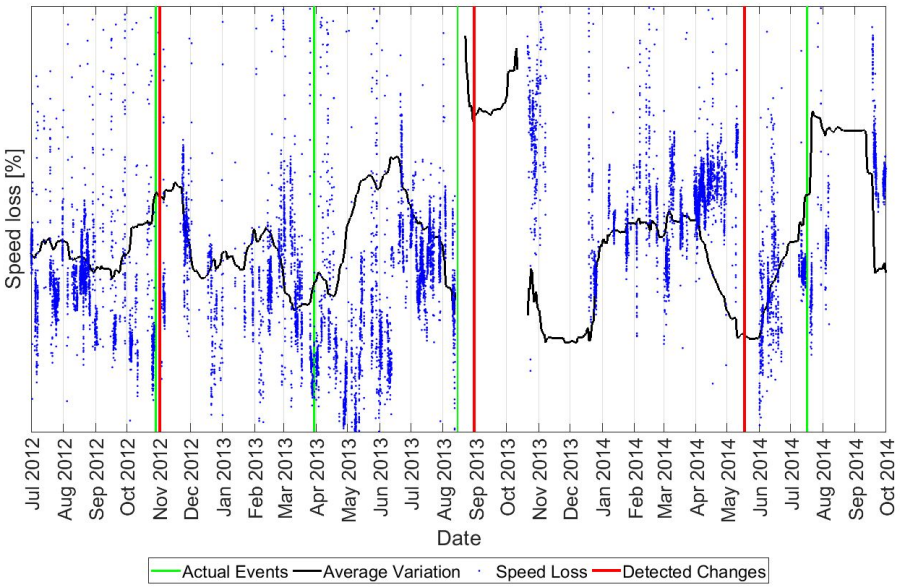


(a) ISO 19030

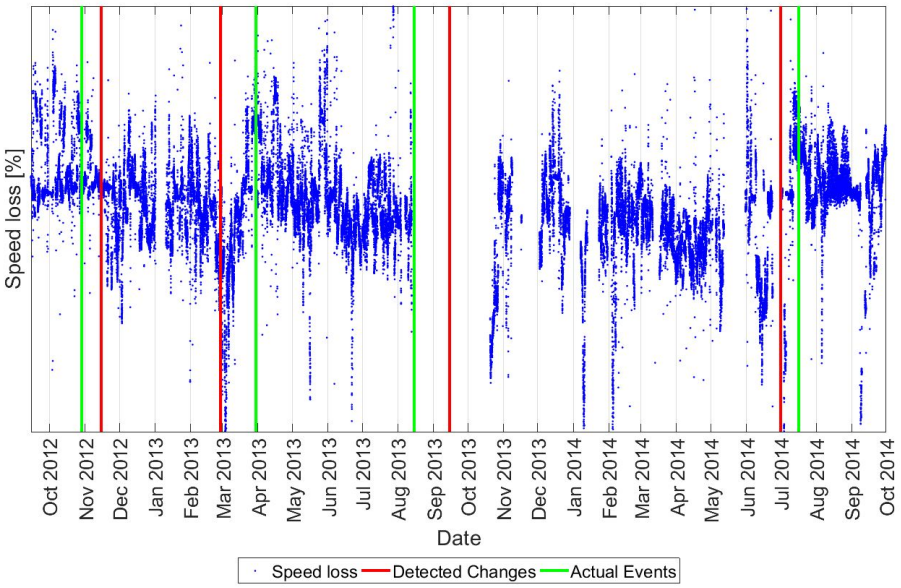


(b) DELM

Figure 3.20: Linear Robust Regression on the Speed Loss Percentages between two consecutive Hull and Propeller Cleaning Events for the V2.

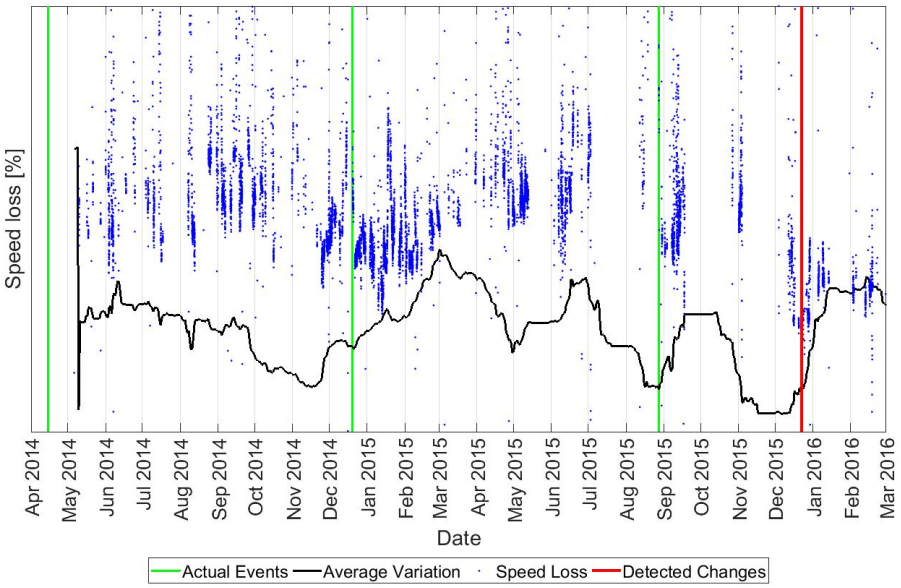


(a) ISO 19030

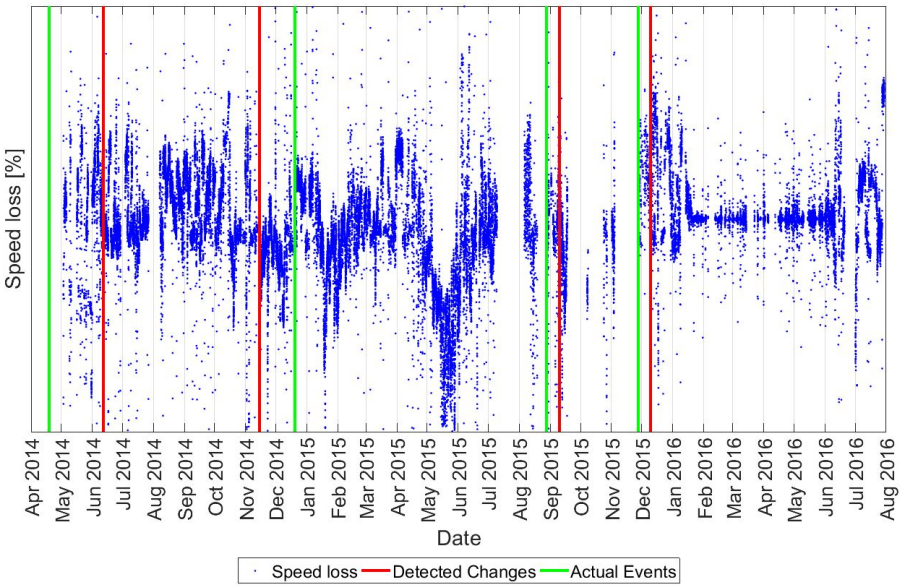


(b) DELM

Figure 3.21: Changes in time of the distribution of the percentage of speed loss estimated with the ISO 19030 and the DELM for the V1.



(a) ISO 19030



(b) DELM

Figure 3.22: Changes in time of the distribution of the percentage of speed loss estimated with the ISO 19030 and the DELM for the V2.

V2 (second maintenance event) only decreased the effect of fouling by a limited extent, while the simultaneous hull and propeller cleaning performed a few months later (third maintenance event) appeared to have a much larger effect. Similarly, the accurate estimations provided by this method could be used to evaluate the efficiency of anti-fouling paints, a widely-adopted solution to reduce the effect of fouling that, however, constitutes at the same time a cost for the company, and has a strong negative impact on the marine environment [ABLWB16]. Given these premises, the ISO standard related to the estimation of marine fouling [ISO15] should be integrated with the proposed methodology which can lead to better results compared to the current methods. If widely adopted and associated to cleaning optimization schedules, it is possible to believe that the proposed method could significantly contribute to an increase in the operational efficiency of the global fleet, hence leading to a reduction in CO_2 emissions from shipping.

3.4 Crash Stop Maneuvering Performance Prediction: a Data Driven Solution for Safety and Collision Avoidance.

The continuous increase of marine traffic and the entry into market of autonomous ships is urging an improvement in safety measures that will guarantee the avoidance of collisions with other moving obstacles at sea. As the majority of collision accidents are caused by human error, an increase in automation and implementation of collision avoidance algorithms is becoming necessary. This rise in automated manoeuvrability requires gaining further insight in the vessel's behaviour. The ship's design has to ensure that the vessel is controllable and capable of manoeuvring securely, even at critical operating conditions.

Crash stop manoeuvring performance is one of the key indicators of the vessel's safety properties for designers and shipbuilders. The crash stop is the maneuver usually performed in critical operating conditions in order to avoid a collision or crashing of a ship into another ship or structure. During this maneuver, the main engine, shaft and propeller is subjected to severe stress and loading since it involves slowing, stopping and reversing the direction as fast as possible. Many factors affect this performance, from the hull design to the environmental conditions, hence it is trivial to assess them accurately during the preliminary design stages.

Several first principal equation methods are available to estimate the crash stop manoeuvring performance [LLYB13, MLY13, OTNS74, Har76, Wir12], but unfortunately, these methods usually are either too costly [LLYB13, Wir12] or not accurate enough [MLY13]. To increase their accuracy, several parameters need to be provided by the various manufacturers of the vessel components, and finally, models need to be fine-tuned based on the outcome of sea trials. All above make the process costly, time-consuming, and not applicable at design stage [LLYB13]. Moreover, suppliers are usually not willing to share technical details which may negatively affect their competitive industrial processes. To overcome these limitations, a new fully DDM was proposed based on RF reported in Section 2.4.3, first developed in [Bre01] and then recently improved

in [BF15], for predicting the crash stopping performance. In this work, the attention was focused on predicting accurately and with minimal computational effort the crash stop characteristics of a vessel in the design stage, for the preliminary assessment of safety requirements imposed by the classification societies [Hop05]. By carefully tuning the RF hyperparameters [OOA16] and by assessing the performance of the final learned model with state-of-the-art resampling techniques [AGOR12], it is shown the effectiveness of the proposal.

In summary, the work contribution is twofold. From a marine engineering perspective, the work deals with the problem of the prediction of the crash stop main characteristics without taking into account the physical laws that are governing the phenomenon. The proposal does not require any a-priory knowledge about the problem and allows exploiting information sources which cannot be modelled with conventional approaches. From a DA perspective, this work proposes an alternative RF formulation and shows that a careful tuning procedure of the RF hyperparameters can remarkably improve its performance. Results from real-world data coming from full scale measurements provided by DAMEN Shipyards, demonstrate the effectiveness of the proposal. In particular, DAMEN, in its many years of ship design and fabrication, has conducted numerous sea trials to measure vessels' general seaworthiness and performance. For this application, a particular cluster of DAMEN vessels, the High-Speed Craft family [3], was used as a test case. The crash stop prediction model of the said vessel can be later on utilized in combination with collision avoidance algorithms.

3.4.1 Adopted Data Cluster

The crash stop is a manoeuvre that is performed when the ship has to achieve an immediate stop, in an emergency situation. The vessel can be stopped by reversing the rotational direction of the Main Engine and thereby the propeller. When the vessel is heading forward, the procedure is to give an order to the engine for full astern, while the rudder is kept in the middle (zero) position, to stop the ship within the minimum interval and shortest possible distance. Such sudden response reduces the speed of the ship, stops her forward motion and after starts heading in the opposite direction from the collision course. This action is implemented during sea trials to assess the safety requirements from the regulations [Int02a, Int02b].

As reported in [Ame06] the stopping ability of a vessel is measured by three main parameters: the Track Reach (TR), the Head Reach (HR) and the Time for Full Maneuver (or time to dead in water) TFM. Also, the Lateral Deviations (LD), Lateral Deviation Direction (LDD), and Heading Deviation Direction (HDD) are parameters of interest, but they are more sensitive to initial conditions and wind disturbances. The crash stop maneuver consists of a stop engine full astern performed after a steady approach at the test speed until the vessel starts going backwards. TR is defined as a distance along the vessel track that the vessel covers from the moment that the

[3] `http://products.damen.com/en/clusters/crew-supply-vessel`

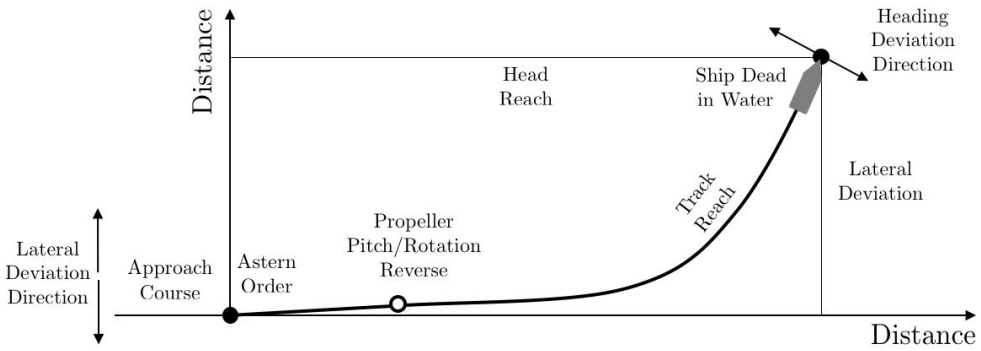


Figure 3.23: Crash Stop maneuver performance indexes.

full astern command is given until ahead speed changes sign. The HR, instead, is the distance along the direction of the course at the moment when the full astern command is given. The distance is measured from the moment when the full astern command is given until the vessel is stopped dead in the water. The LD is defined as the distance perpendicular to the direction of the course at the moment when the full astern command is given. Also, this distance is measured from the moment when the full astern command is given until the vessel is stopped dead in the water. Figure 3.23 shows the meaning of each parameter. At the design stage, in order to assess the maneuvering characteristics both in the trial and full load conditions, reliable methods should be applied. These methods should ensure satisfactory accuracy for the prediction of new vessels'response and satisfactory extrapolation of trial results for the full load condition. As reported in [Har76], the factors which affect the stopping ability of vessels are the displacement, the initial speed, the block coefficient, the vessel hull fouling degree, the main engine full astern power, the time taken to effect changes in engine telegraph settings, the propeller category, and the environmental conditions (e.g. wind, stream, and the depth of water). During the maneuver, the interaction between hull and propeller(s) is quite complex to model. For this reason, empirical calculations are used since appropriate motion equation coefficients are not available [LLYB13, MLY13].

In this work, the attention was focused on the crash stop from maximum operational speed, as the worst case scenario, and any transient mode speed according to the available data provided by DAMEN Shipyards.

To prove the effectiveness of the new data-driven approach, the attention focused on a particular cluster of DAMEN vessels, the High-Speed Craft (HSC) family, but the method is general, and the data can be easily retrieved by any ship type. The total amount of the vessels is 230 divided into four product clusters: Fast Crew Supplier, Search And Rescue, Stan Pilot and Stan Tender. For each product cluster, different sub-products (vessel type) are available, each one consisting of several yard numbers. Every yard number has performed and reported several crash stop maneuvers. For each of the vessels, the information reported in Table 3.16 is available. During

Table 3.16: Available vessels information.

Variable name	Unit	Variable name	Unit
Product Cluster	[]	Gearbox reduction ratio	[]
Product	[]	Propeller inertia in air	$[\frac{kg}{m^4}]$
Yard Number	[]	Water Density	$[\frac{kg}{m^3}]$
Location	[]	Design vessel speed	[knots]
Country	[]	Design wake factor	[]
Trial Engineer	[]	Design propeller pitch	[]
Orientation	[]	Volume	$[m^3]$
Crash Stop type	[]	Longitudinal center of buoyancy	[m]
Initial vessel speed	[knots]	Transversal center of buoyancy	[m]
Initial heading	[]	Vertical center of buoyancy	[m]
Initial Engine Speed	[rpm]	Vertical center of gravity	[m]
Heading Deviation	[deg]	Waterplane area	$[m^2]$
Propeller mass	[kg]	Waterplane area inertia (x-axis)	$[m^4]$
Under Keel Clearance	[m]	Waterplane area inertia (y-axis)	$[m^4]$
Rotative efficiency (design)	[]	Waterline length	[m]
Wave Height	[m]	Waterline breadth	[m]
Wave Direction	[deg]	Midship section area	$[m^2]$
Wind Velocity	[m/s]	Wetted surface	$[m^2]$
Wind Direction	[bar]	Midship draught	[m]
Current velocity	[knots]	Roll angle	[deg]
Current direction	[deg]	Pitch angle	[deg]
Loading condition	[]	Longitudinal center of floatation	[m]
Draught aftmark	[m]	Transversal center of floatation	[m]
Draught foremark	[m]	Propeller inertia in water	$[\frac{kg}{m^4}]$
Static trim	[m]	Main engine type	[]
Displacement	[tons]	Main engine nominal power	[kW]
Longitudinal center of gravity	[m]	Gearbox manufacturer	[]
Number of driveline	[]	Gearbox type	[]
Propeller diameter	[m]	Main engine nominal speed	[rpm]
Number of propeller blades	[]	Propeller manufacturer	[]
Blade area ratio	[]	Propeller type	[]
Engine break power	[kW]	Propeller diameter	[m]
Engine speed	[rpm]	Propeller number of blades	[]

the trials, the maneuvers are digitally recorded using advanced portable measurement equipment (Differential GPS). Therefore, the goal is to predict, based on the information of Table 3.16, available at design stages, the crash stop maneuver performance indexes (TR, HR, TFM, LD, LDD, and HDD).

3.4.2 Extrapolation Results

In this section, the results of the application of the different ML techniques are reported. In particular, three approaches reported in Section 2.4.3 have been compared:

- ORF: the original RF proposed in [Bre01];

- RFR: the RF proposed in [BF15] which improve over the ORF;
- PRF: the RF algorithms of Algorithm 1 where their hyperparameters have been tuned with the BTS procedure described in Section 2.4.3.

For what concerns PRF, $\mathcal{H}$ is set to $\{\gamma, b, n_v, n_r, n_l\}$ and $\mathfrak{H} = \{10^{-4.0}, 10^{-3.5}, \cdots, 10^{3.0}\} \times \{0.7, 0.8, \cdots, 1.2\} \times d^{\{0.0,0.1,\cdots,1\}} \times \{1, 10, 100\} \times n \cdot \{0.0, 0.01, 0.05, 0.1\}$, with $n_o = 100$ in the BTS procedure, and $n_t = 10^3$ since larger values did not produce any improvement in the accuracy of ORF, RFR, and PRF in any of the experiments.

Moreover, three different scenarios have been investigated:

- S1: different yard number are kept in each of the sets $\mathcal{L}_l^o$, $\mathcal{V}_v^o$, and $\mathcal{T}_t^o$. In this way in the training set, both examples of different products and different product clusters are present;
- S2: different products are kept in each of the sets $\mathcal{L}_l^o$, $\mathcal{V}_v^o$, and $\mathcal{T}_t^o$. In this way the case when a new product needs to be designed is simulated;
- S3: different product clusters are kept in each of the sets $\mathcal{L}_l^o$, $\mathcal{V}_v^o$, and $\mathcal{T}_t^o$. In this way, the case when a new series of products needs to be designed is simulated.

Note that S1 is a simpler task with respect to S2, which is again a simpler task comparing to S3. In fact, it is simulated the increasingly difficult task to extrapolate the performance indexes of a vessel, which is more and more different with respect to the vessels contained in the training set.

Table 3.17 reports $\widehat{L}(\mathscr{A}_{\mathcal{H}^*}(\mathcal{D}_n))$ and $L(\mathscr{A}_{\mathcal{H}^*}(\mathcal{D}_n))$ in percentage respectively for ORF, RFR, and PRF in S1, S2, and S3 where $\delta = 0.05$. From the results it is possible to observe that:

- PRF mostly outperform ORF and RFR as expected;
- $\widehat{L}(\mathscr{A}_{\mathcal{H}^*}(\mathcal{D}_n))$ and $L(\mathscr{A}_{\mathcal{H}^*}(\mathcal{D}_n))$ are close to each other, and this means that it is possible to guarantee a quality of the estimation which is close enough to the expected quality of the produced DDM;
- as expected, the performances in S1 are better than the ones in S2 and S3. Nevertheless, even in S2 an S3 the performances of PRF are quite satisfying since the errors are around 5%. Note, instead, that for S3, ORF and RFR cannot be used in a real-world application because of their low accuracy.

Moreover, in Table 3.18, the selected hyperparameters by the BTS for PRF in S1, S2, S3 are reported. From the table it is possible to observe that:

- when the task is simple (S1 and S2) and PRF do not perform very much differently from ORF and RFR, the hyperparamenters selected are very close to the one of ORF and RFR;
- when the task is harder (S2 and S3) and PRF works much better than ORF and RFR it is possible to see also that the hyperparameters selected by the BTS are very far from the ones of ORF and RFR and this supports the previous statements.

In Figures 3.24, 3.25, and 3.26, the ten most relevant features for the problem are reported, adopting PRF as FR method, for the six output features and the three problems S1, S2, and

Table 3.17: $\widehat{L}(\mathscr{A}_{\mathcal{H}^*}(\mathcal{D}_n))$ and $L(\mathscr{A}_{\mathcal{H}^*}(\mathcal{D}_n))$ of ORF, RFR, PRF in S1, S2, S3 (in %).

		$\widehat{L}(\mathscr{A}_{\mathcal{H}^*}(\mathcal{D}_n))$			$L(\mathscr{A}_{\mathcal{H}^*}(\mathcal{D}_n))$		
Loss Function		S1 ORF	S1 RFR	S1 PRF	S1 ORF	S1 RFR	S1 PRF
TR	ℓ_{TRAE}	3.9±0.4	3.1±0.3	2.7±0.3	7.1±0.7	5.8±0.5	5.2±0.6
HR	ℓ_{TRAE}	3.7±0.4	3.0±0.3	2.7±0.3	6.5±0.7	5.8±0.6	5.2±0.4
LD	ℓ_{TRAE}	30.8±3.2	12.2±1.1	2.9±0.3	36.8±3.7	16.5±1.7	5.8±0.5
TFM	ℓ_{TRAE}	3.8±0.4	3.1±0.3	2.7±0.3	7.1±0.6	5.8±0.6	5.2±0.5
LDD	ℓ_{H}	7.1±0.8	5.7±0.6	4.3±0.5	10.7±1.2	8.9±0.8	7.7±0.7
HDD	ℓ_{H}	8.2±0.9	5.7±0.6	4.1±0.4	11.9±1.3	8.9±0.9	7.1±0.7
Loss Function		S2 ORF	S2 RFR	S2 PRF	S2 ORF	S2 RFR	S2 PRF
TR	ℓ_{TRAE}	10.2±1.0	4.0±0.4	3.1±0.4	14.2±1.3	7.1±0.7	5.8±0.6
HR	ℓ_{TRAE}	10.8±1.1	4.3±0.4	3.3±0.3	15.3±1.7	7.7±0.8	6.5±0.6
LD	ℓ_{TRAE}	37.1±3.9	14.9±1.6	5.0±0.5	43.0±3.7	19.8±2.0	8.3±0.9
TFM	ℓ_{TRAE}	12.2±1.2	4.9±0.5	4.9±0.5	16.5±1.6	8.3±1.1	8.3±0.9
LDD	ℓ_{H}	10.3±1.1	6.2±0.7	5.3±0.6	14.8±1.5	9.5±0.9	8.9±1.0
HDD	ℓ_{H}	11.7±1.2	7.1±0.7	4.7±0.4	15.9±1.7	10.7±1.0	7.7±0.7
Loss Function		S3 ORF	S3 RFR	S3 PRF	S3 ORF	S3 RFR	S3 PRF
TR	ℓ_{TRAE}	12.9±1.4	3.9±0.4	3.7±0.4	17.6±1.7	7.1±0.7	6.5±0.6
HR	ℓ_{TRAE}	13.3±1.1	4.0±0.4	4.0±0.4	18.1±1.9	7.1±0.8	7.1±0.8
LD	ℓ_{TRAE}	36.1±3.7	14.5±1.4	5.4±0.5	42.0±4.6	19.2±1.9	8.9±0.9
TFM	ℓ_{TRAE}	12.6±1.2	5.0±0.5	5.0±0.5	17.0±1.8	8.3±0.9	8.3±0.8
LDD	ℓ_{H}	26.7±3.1	8.0±0.7	3.9±0.4	32.1±3.6	11.9±1.3	7.1±0.8
HDD	ℓ_{H}	25.6±2.4	7.7±0.9	3.9±0.4	31.1±3.7	11.3±1.2	7.1±0.8

Table 3.18: Selected hyperparameters by the BTS for PRF in S1, S2, S3.

	S1					S2					S3				
	γ	b	n_v	n_r	n_l	γ	b	n_v	n_r	n_l	γ	b	n_v	n_r	n_l
TR	$10^{0.5}$	1.1	$d^{0.5}$	10	$n{\cdot}0.0$	$10^{0.5}$	1.0	$d^{0.7}$	10	$n{\cdot}0.1$	$10^{0.0}$	0.9	$d^{0.7}$	100	$n{\cdot}0.1$
HR	$10^{0.0}$	1.0	$d^{0.6}$	10	$n{\cdot}0.0$	$10^{0.0}$	0.9	$d^{0.6}$	10	$n{\cdot}0.1$	$10^{0.0}$	0.9	$d^{0.7}$	100	$n{\cdot}0.1$
LD	$10^{0.5}$	1.1	$d^{0.6}$	10	$n{\cdot}0.0$	$10^{0.0}$	1.0	$d^{0.6}$	100	$n{\cdot}0.1$	$10^{-0.5}$	1.0	$d^{0.7}$	100	$n{\cdot}0.1$
TFM	$10^{1.0}$	1.2	$d^{0.5}$	1	$n{\cdot}0.0$	$10^{0.5}$	1.1	$d^{0.6}$	10	$n{\cdot}0.1$	$10^{-0.5}$	1.0	$d^{0.7}$	10	$n{\cdot}0.1$
LDD	$10^{1.0}$	1.1	$d^{0.6}$	10	$n{\cdot}0.1$	$10^{0.5}$	1.0	$d^{0.7}$	100	$n{\cdot}0.1$	$10^{-0.5}$	0.9	$d^{0.8}$	100	$n{\cdot}0.2$
HDD	$10^{1.0}$	1.1	$d^{0.5}$	10	$n{\cdot}0.1$	$10^{0.5}$	1.0	$d^{0.6}$	10	$n{\cdot}0.1$	$10^{-0.5}$	1.0	$d^{0.7}$	100	$n{\cdot}0.2$

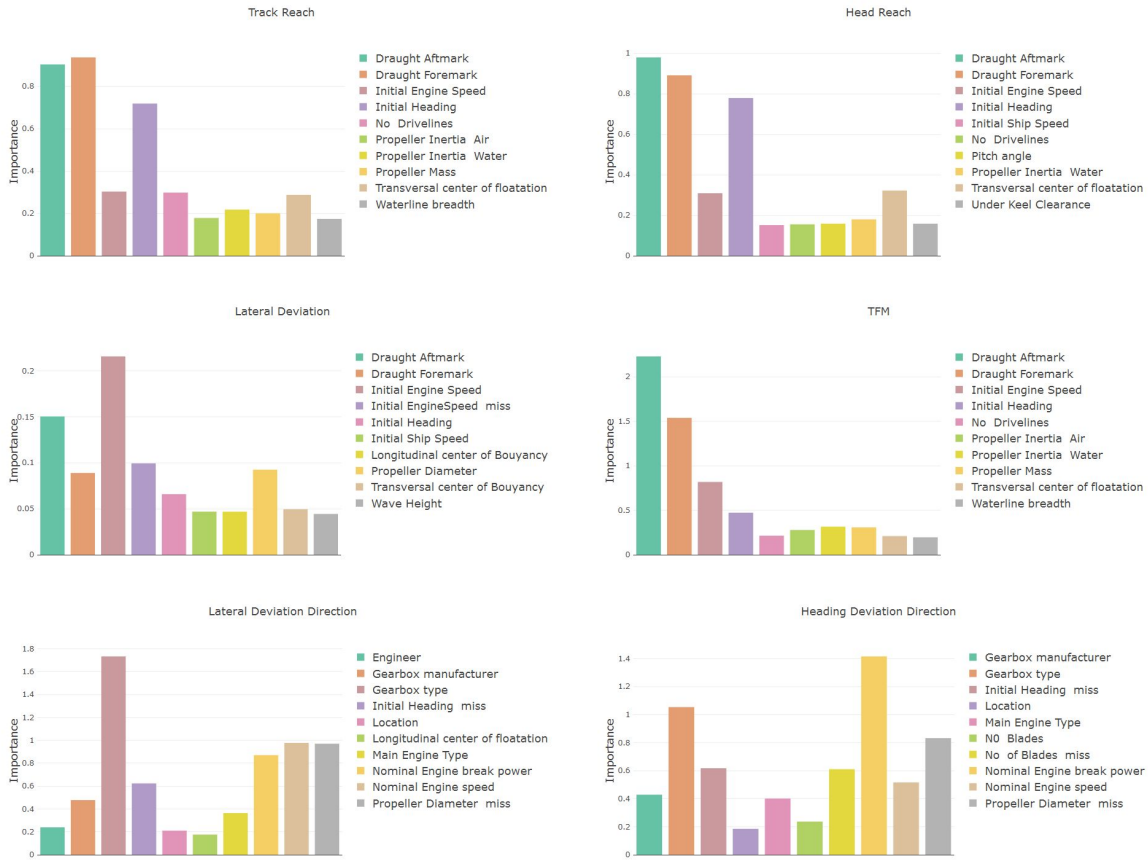


Figure 3.24: Top 10 Feature Importance computed with S1.

S3. From the Figures it is possible to note that the model can adequately account for the relevant features related to TR, HR, TFM, and LD. The first three features are consistent with the different scenarios and in line with the available knowledge of the phenomena. As expected, the draughts, the initial engine speed and the initial vessel speed have high predictive power.

When it comes to LDD and HDD, the hydrodynamic unbalances acting on the hull during the maneuver are dominating the behavior of the vessel. Although when discussing vessels operating in confined waters or inland shipping, these parameters are of some interests are very sensitive to the initial conditions and weather disturbances. It is therefore noted that the outcome of the sea trials is highly influenced by exogenous factors such as a pre-existing initial rate of turn, the wind and wave direction and the rudder position. Finally, it is important to remark that due to the high sensitivity of the LDD and HDD features, the exact ranking values are highly dependent on the different localized attributes, thus making the generalization process of the FR a difficult task to interpret.

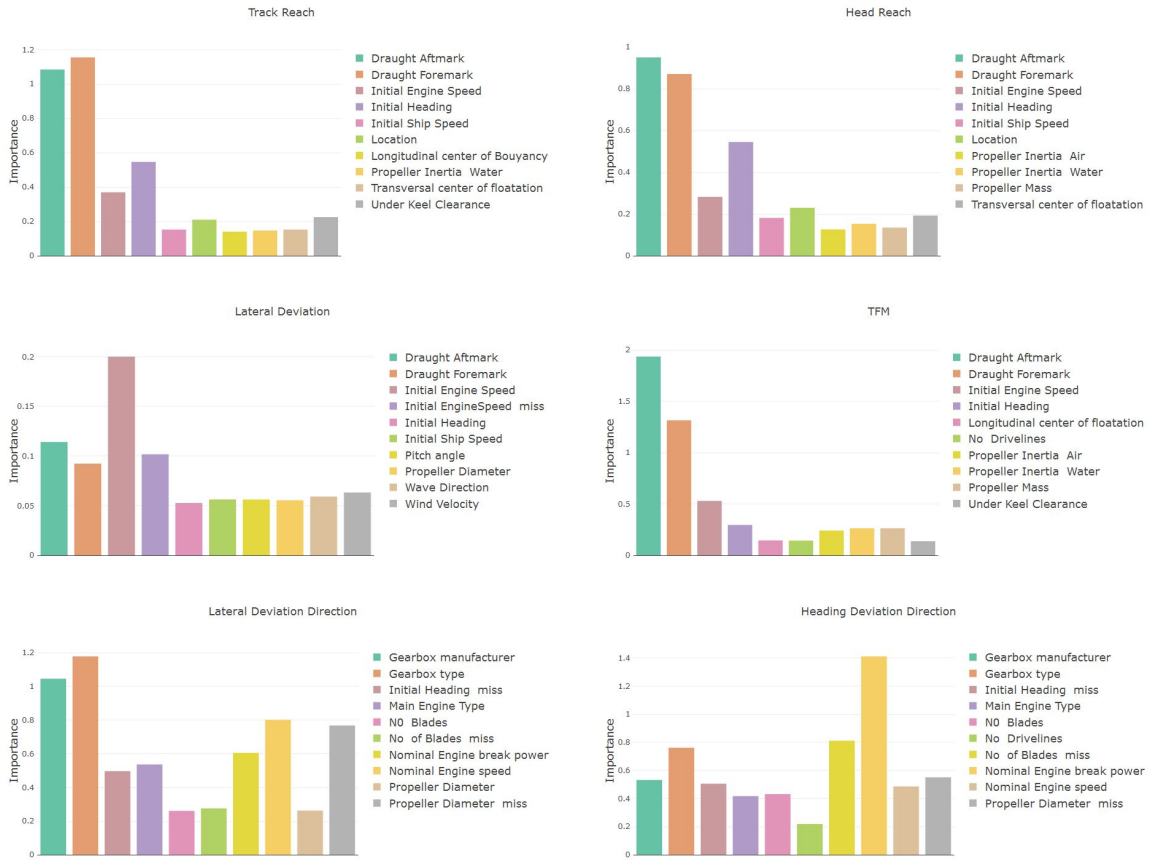


Figure 3.25: Top 10 Feature Importance computed with S2.

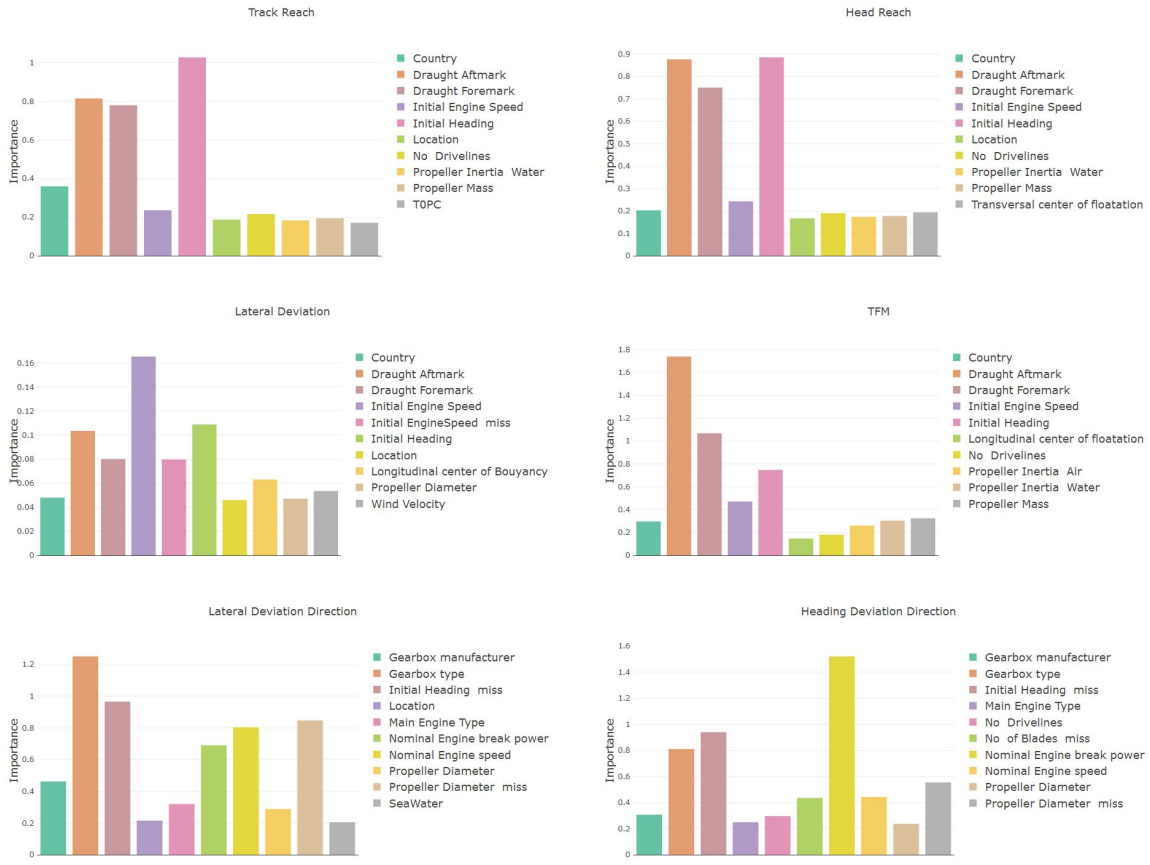


Figure 3.26: Top 10 Feature Importance computed with S3.

3.5 Model Scale Cavitation Noise Spectra Modelling: Combining Physical Knowledge with Data Science

In the latest years ships Underwater Radiated Noise (URN) has become subject of great interest because of the increasing attention to the environmental impact of human activities of [Dir08] and the improved sensibility to the on-board comfort [IMO12]. In this context, much effort has been spent in the study of propeller noise since it represents the dominating noise source on ships, especially when it cavitates [Car12].

The prediction of propeller cavitation can be addressed by means of different approaches, as summarized in Figure 3.27. The approach based on CFD models [BB10b, BB10a] is able to predict the full scale noise knowing hull and propeller geometry, and ship's operating conditions. CFD-based models, even if showing promising capabilities, are still under development and their high computational requirements limit their use to the field of research, being it impractical in a normal propeller design loop.

The second approach is the adoption of Model Scale Tests (MST), which are mainly carried out in cavitation tunnels; this approach is still largely considered the most reliable method for cavitation noise prediction. MST makes use of a model of the propeller (realized knowing the geometry from the design) which is tested reproducing cavitation conditions in order to measure the model scale noise. Scale effects must be carefully taken into account in order to retrieve the full scale noise. In particular, the development of Tip Vortex Cavitation (TVC) is significantly influenced by the Reynolds number, as remarked by [McC62]: the onset of TVC in model scale occurs at significantly lower cavitation number than on the full scale propeller. As a consequence, for some ship operational conditions it is not trivial to correctly reproduce the cavitation pattern in MST. Moreover, MST are quite expensive and time-consuming; it is not feasible to include them in the early stage of the design.

On the other hand, MSTs provide the opportunity to collect a large amount of data suitable for the definition of a cavitation noise model which can replace an actual MST in the cavitation tunnel.

In order to avoid time-consuming MSTs, it is possible to replace them with a model able to predict the propeller noise by having only the information about the propeller geometry, the inflow wake and cavitation behaviour; these characteristics may be predicted (e.g. by means of CFD calculations) without MSTs. These models can be built with different approaches.

The most common one is to use semi-empirical formulations to build simplified PMs combined with empirical relations defined from experimental data. PMs correlate noise spectra with the main characteristics of propeller geometry, operational conditions, cavitation patterns and generally those data already available at the design stage of the propeller. Examples of such approaches in the study of propeller cavitation noise are represented by the work of [Rae96] and [Bos18].

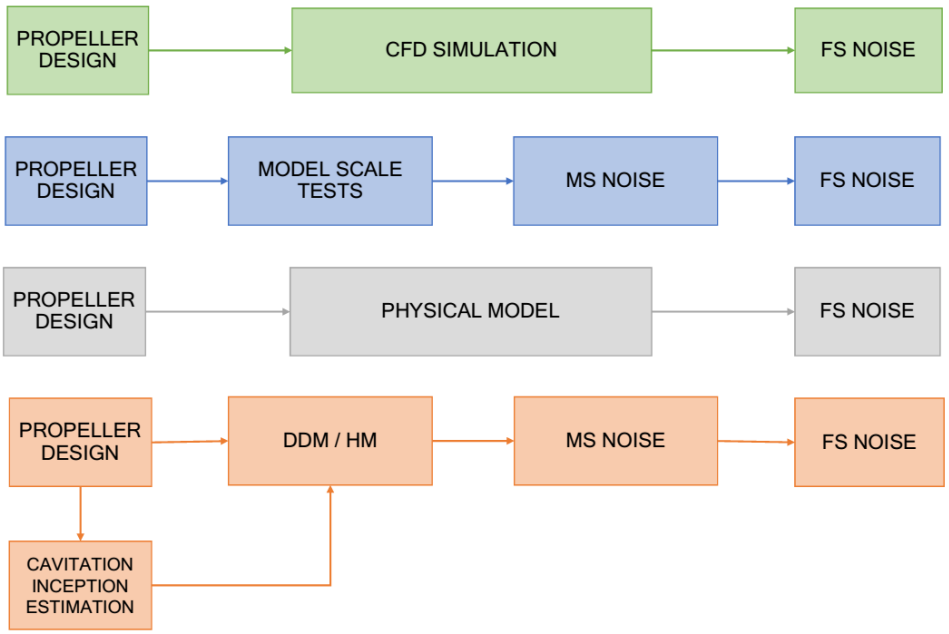


Figure 3.27: Cavitation noise prediction methodologies: CFD, MST, PM, DDM, and HM.

The limitation of these PMs lies in the fact that it is not trivial to model all the phenomena which influence the cavitation noise and, consequently, the quality of the results may not be completely satisfactory requiring an ad hoc tuning.

For this reason, in this work DDMs were employed, which are able to build models exploiting robust statistical inference procedures and data collected in previous MSTs in order to make predictions about previously unseen MSTs. However, DDMs usually produce black-box (non-parametric) models that are not supported by any physical interpretation; this, despite representing a possible advantage, as mentioned above, may limit the capability of the models themselves, without exploiting important knowledge about the phenomena of interest. Moreover, in general, a great amount of historical data is necessary in order to build reliable models.

For this reason, in this work a hybrid approach is proposed, namely HMs, in order to take advantage of the best characteristics of both PMs and DDMs by combining them together, as described in Section 2.6. HMs are widely used in those contexts were the experience on the field brought by PMs can enhance the DDMs prediction [COBA17].

In order to develop and test the models proposed in this work, a dataset was first collected by means of an extensive set of cavitation tunnel MSTs of two controllable pitch propellers of twin screw ships. Then the propeller URN has been parameterized with a simple procedure which allows to extract its main characteristics.

The modelization of only the continuous part of the cavitation noise spectrum is addressed in this

work.

Then different models (PMs, DDMs, and HMs) have been developed for predicting these main characteristics based on quantities that can be estimated without performing any MST. This is done with the aim, in the future, to obtain a prediction tool for the estimation of the propeller noise, once a larger set of data will be available. In parallel to this, the same tool may be adopted in order to predict propeller noise in correspondence to conditions which cannot be reproduced in model scale. With these two objectives, the models have been validated with the real data in two different scenarios:

- Interpolation Scenario: in this case models try to predict the propeller noise spectra main characteristics in various, but different, working conditions within the ones exploited for building the model;
- Extrapolation Scenario: in this case models try to predict the propeller noise spectra main characteristics in groups of working conditions where the cavitation intensity is different w.r.t. the one exploited for building the model. These extrapolation tests are useful to test the capability of the models to predict the noise related to cavitation patterns which cannot be correctly reproduced at model scale.

Results show that the proposed HMs accuracies are remarkably higher than the ones of PMs and DDMs, both in interpolation and extrapolation scenarios.

Due to confidentiality issues, sensitive data like propeller geometry and working conditions are omitted or altered by means of appropriate reference values kept constant for the whole article.

3.5.1 Experimental Campaign

In this section a general description of the MST performed in the cavitation tunnel and the collected data is reported. Experiments have been performed at the cavitation tunnel of the University of Genoa; for a detailed description of the facility see [TAVA17]. MST have been performed for a total of 425 propeller loading conditions chosen in order to properly explore the variables'domain. All MST have been performed on two controllable pitch propellers, respectively referred as Propeller 1 (P1) and Propeller 2 (P2), tested at various pitch settings. The main characteristics are reported in Table 3.19.

For each propeller, a complete set of tests has been performed: cavitation bucket, cavitation observations, pressure pulses and propeller radiated noise measurements in a large set of operational conditions. All these measurements provide detailed information about cavitation typologies present and their extensions for a wide range of operational conditions to allow correlating measured noise with these values.

Radiated noise measurements were carried out with two hydrophones, respectively referred as H1 and H2. H1 is located inside a tank full of water attached to the Plexiglas window under

Table 3.19: The P1 and the P2 model characteristics.

Variable	P1	P2
Number of blades	5	5
Diameter	$0.25m$	$0.25m$
Direction of rotation	Clockwise	Clockwise
Design pitch ratio at 0.7R	1.385	1.156
Reduced (-3°) pitch ratio at 0.7R	1.229	1.013
Reduced (-5°) pitch ratio at 0.7R	-	0.938
Reduced (-6°) pitch ratio at 0.7R	1.082	-
Incremented (+2°) pitch ratio at 0.7R	-	1.256
Shaft inclination	6.8°	2.5°

the propeller. H2 is mounted on a fin immersed in the tunnel flow, outside of the propeller slipstream: only data acquired by the latter have been used in this work because of the better signal-to-noise ratio. The propeller loading conditions for model tests are defined according to the identity of the thrust coefficient K_T and the cavitation number based on rotational speed σ_n. The first coefficient represents the kinematic condition of the propeller while the cavitation number defines a cavitation similarity criterion.

Noise measurements have been performed mainly following the International Towing Tank Conference (ITTC) model scale noise measurement guidelines [ITT17]. Data used for the following analyses are spectra of the Radiated Noise Levels (RNL) at the distance of $1m$ from the propeller center disk, obtained after having applied the background noise correction. The transfer function correction for the confined environment effect [TVAN16, BFF13], is identical for all the noise measurements considered in present work, hence it has not been applied here since it does not affect the analysis of the modelling approaches considered. Thus, the developed models describe the characteristics of noise spectra measured inside the cavitation tunnel; in order to compute free field noise spectra, transfer function corrections could be applied to noise spectra predicted by the models. The application of transfer functions to noise spectra used in building the model is needed when considering results of tests carried out with different configurations or from different institutes (e.g. [AAFS18]), activities which may represent possible future improvement of this work. As a consequence, transfer functions will be considered for future activities.

The working conditions, for which noise samples are collected, have been chosen in order to provide an exhaustive characterization of cavitation noise. The characterization of propeller cavitation is represented by the cavitation bucket. In Figure 3.28 the cavitation bucket for the P2 at design pitch is reported as an example, while others are not presented for the sake of shortness; nevertheless, the following considerations are valid for the whole data collection. The cavitation bucket reports the inception of different cavitation phenomena as function of the thrust coefficient and cavitation number.

Quantities collected for the working points defined in 3.28 are summarized in Table 3.20. These values represent the features considered as possible input for the model developed as described.

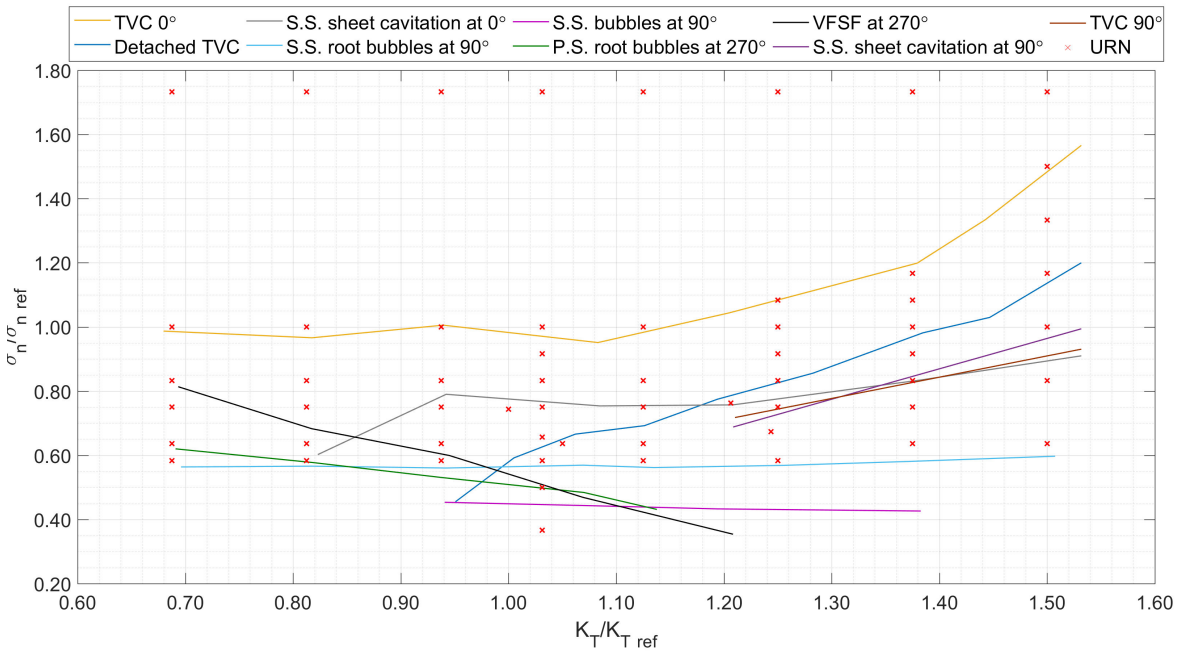


Figure 3.28: P2 design pitch cavitation bucket: the inception of the different cavitation phenomena is reported as function of the thrust coefficient ($^{K_T}/_{K_{Tref}}$) and the cavitation number ($^{\sigma_n}/_{\sigma_{nref}}$).

Table 3.20: Dataset input variables.

Propeller working parameters

Variable	Unit	Description
$^{P}/_{D}$	[]	Pitch ratio
$\Delta\Phi$	[°]	Difference between actual and design pitch
J	[]	Advance coefficient
K_T	[]	Thrust coefficient
$10K_Q$	[]	Torque coefficient
η_o	[]	Open water efficiency
σ_v	[]	Cavitation index based on advance velocity
σ_n	[]	Cavitation index based on rotational speed
Va	[m/s]	Advance velocity
n	[rps]	Propeller rotation
Tc	[kgf]	Thrust
Qc	[kgf·cm]	Torque
p_{rel}	[mBar]	Pressure relative to the ambient pressure

Cavitation types

Variable	Unit	Description
TVC 90°	[]	Suction side tip vortex
D. TVC	[]	Detached tip vortex
TVC 0°	[]	Suction side tip vortex at 0°
S.S. S	[]	Suction side sheet
S.S. S 0°	[]	Suction side sheet at 0°
S.S. RB	[]	Suction side root bubbles
S.S. B	[]	Suction side bubbles
VFSF	[]	Vortex from sheet face
P.S.TVC	[]	Pressure side tip vortex
P.S. S	[]	Pressure side sheet
P.S. RB	[]	Pressure side root bubbles

Wake parameters

Variable	Unit	Description
Wwd_{07}	[°]	Wake width at 0.7R
$D_\theta W\|_{07}^{-}$	[°]	Left wake gradient at 0.7R
$D_\theta W\|_{07}^{+}$	[°]	Right wake gradient at 0.7R
Wwd_{09}	[°]	Wake width at 0.9R
$D_\theta W\|_{09}^{-}$	[°]	Left wake gradient at 0.9R
$D_\theta W\|_{09}^{+}$	[°]	Right wake gradient at 0.9R

Angle of attack geometric

Variable	Unit	Description
$\overline{\alpha}_{G07}$	[°]	Circumferential average α_G at 0.7R
min α_{G07}	[°]	Minimum α_G at 0.7R
max α_{G07}	[°]	Maximum α_G at 0.7R
$\theta\|_{max\ \alpha_{G07}}$	[°]	Angular position of maximum α_G at 0.7R
$\overline{\alpha}_{G09}$	[°]	Circumferential average α_G at 0.9R
min α_{G09}	[°]	Minimum α_G at 0.9R
max α_{G09}	[°]	Maximum α_G at 0.9R
$\theta\|_{max\ \alpha_{G09}}$	[°]	Angular position of maximum α_G at 0.9R

The features include some characteristics of the propeller geometry, the characteristics of propeller inflow and the propeller working conditions in terms of kinematic conditions, propeller loading and cavitation. All these quantities are usually available at the propeller design phase, exception made for the inception indices which may be estimated anyway by dedicated CFD simulations with reasonable accuracy, see [GTVC14] as an example. It has to be remarked that the CFD simulations to obtain the inception indices are much more accurate and computationally inexpensive than those required to estimate directly the model scale noise from the propeller geometry. As shown in Figure 3.27, the output data of the system built in the present work is represented by the spectra of measured propeller noise, and in particular the continuous part of the spectrum associated with cavitation noise. Spectra are usually analyzed in one third octave band or in narrowband representation; however, the trends featured by single spectral rows may be rather complex and not so meaningful, especially if compared to the general trends observed

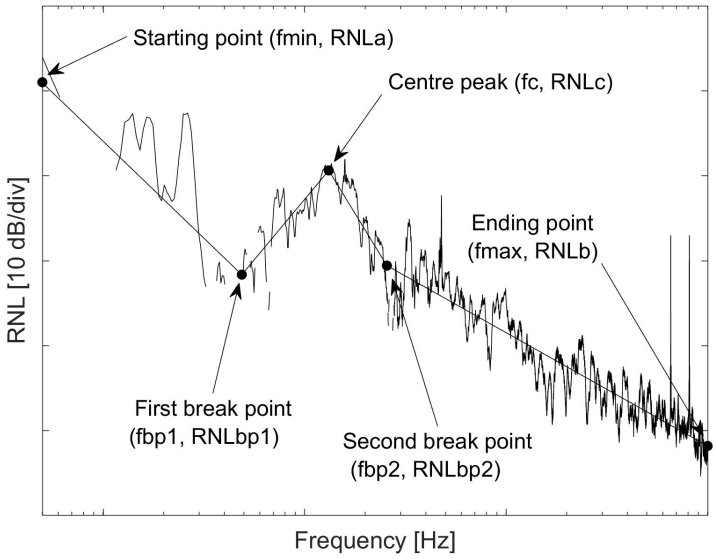


Figure 3.29: Adopted spectrum simplification.

Table 3.21: Simplified noise spectrum variables.

URN

Frequencies			RNL		
Variable	Unit	Description	Variable	Unit	Description
f_{min}	$[\log_{10}(\text{Hz})]$	Starting frequency	RNL_{a}	[dB]	Noise level at f_{min}
f_{bp1}	$[\log_{10}(\text{Hz})]$	Frequency at first breakpoint	RNL_{bp1}	[dB]	Noise level at f_{bp1}
f_{c}	$[\log_{10}(\text{Hz})]$	Central frequency	RNL_{c}	[dB]	Noise level at f_{c}
f_{bp2}	$[\log_{10}(\text{Hz})]$	Frequency at second breakpoint	RNL_{bp2}	[dB]	Noise level at f_{bp2}
f_{max}	$[\log_{10}(\text{Hz})]$	Ending frequency	RNL_{b}	[dB]	Noise level at f_{max}

in the experiments. In addition, modelling the complete spectra would increase significantly the computational time.

Due to this a simplified description of the spectra is defined, keeping only the information of physical and practical relevance. The adopted simplification, shown in Figure 3.29, reproduces the spectral shapes typical of cavitation noise which are described.

This simplified spectrum is defined by the knowledge of the frequency and level of only five points; these frequencies and levels, summarized also in Table 3.21, are the target of the developed models. Target frequencies are expressed in terms of logarithm of the frequency to keep the logarithmic sensitivity of noise to frequency. The first portion of the spectrum, from the starting point to the first break point, is dominated by background noise or by tonal noise components, hence not of interest in the present work. The part of the simplified spectrum which can be

Table 3.22: PMs estimated parameters.

Propeller	$\Delta\Phi$	$\tau/\overline{\tau}$	$a_p/\overline{a_p}$
P1	0	1.006	0.991
P1	-3	0.964	0.980
P1	-6	0.691	1.018
P2	0	1.077	1.019
P2	+2	1.117	0.989
P2	-3	0.686	1.062
P2	-5	0.648	1.004
P1 & P2	-	1	1

reasonably considered representative of cavitation noise is identified by frequencies higher than those of the first break point, hence the starting point is neglected in the following analyses. Besides, starting and ending frequencies are constant, and they do not need to be modelled. Hence, the models targets are f_{bp1}, RNL_{bp1}, f_{c}, RNL_{c}, f_{bp2}, RNL_{bp2} and RNL_{b}.

3.5.2 Physical Model Equations

In the present work, some physics-based models for f_{c} and RNL_{c} are derived with a twofold objective: provide a benchmark for comparison with DDMs and provide features for the HMs. The tuning of parameters for the PMs is done by fitting on the whole data set, hence no distinction between $\mathcal{D}_n$ and $\mathcal{T}_m$ has been done. The frequency and the maximum level of the peak in the spectrum of tip vortex noise have been computed with an approach similar to the one presented in [Bos09] for the vortex resonance frequency and in [Rae96] for the sound pressure level.

According to the approach used, the vortex strength is assumed to be proportional to the thrust coefficient by means of the coefficient τ which represents the relative tip loading, and it is here assumed to be dependent only on propeller geometry and wake field. The vortex cavitating radius is then computed using the potential vortex model.

By making use of the experimental data presented in [MA97] and formulations based on theoretical considerations, it is possible to derive the resonance frequency of the vortex, as given by Eq. (3.25):

$$f_{\text{c}} = \frac{0.45\pi^2 \sigma_{\text{tip}} n Z}{\tau K_T} \tag{3.25}$$

The cavitation index σ_{tip} is based on the resultant velocity at the blade tip and Z is the blades number. By doing this, the only unknown parameter is the coefficient τ, which can be obtained by means of fitting the data collected in current experiments. Values obtained are reported in Table 3.22. All the values are made non dimensional w.r.t. the value of the parameter tuned on the whole dataset, as explained below more in detail.

As expected the coefficient τ, for a given propeller, decreases while reducing the pitch, indicating the overall reduced load.

For what regards the amplitude of the spectral peak, the prediction is based on [Rae96], and the relation between sound pressure levels and propeller functioning parameters is given by:

$$\mathrm{RNL_c} = a_p + 20\log_{10}\left[\left(\frac{\tau K_T}{Z\sqrt{\sigma_n}}\right)^k \sqrt{Z}\right] \tag{3.26}$$

According to the work of [Rae96], the value of the exponent k should be 2; however, as pointed out also in [Bos18], the fitting with data is improved considering higher values. For the present work, the parameter k has been chosen equal to 3. The value of a_p is instead directly estimated by fitting to experimental data.

When a new propeller is considered, the values of a_p and τ cannot be estimated from data fitting before experimental results are available. Hence, it is important to define procedures for their estimation before experimental tests are carried out. This problem could be overcome using procedures like that proposed in [Bos18], according to which the vortex strength is directly derived by Boundary Element Methods (BEM) computations and consequently the knowledge of τ is no more needed; alternatively, the value of these parameters may be correlated with available geometric or hydrodynamic characteristics of the propellers. Obviously, the latter solution requires the availability of a significant number of test cases. In case this is not possible, the average values of the two coefficients obtained from previous experiments may be used as a first rough estimate. In order to check how the accuracy of the PM is affected by this possible rough assumption, the coefficients have been hence calculated also on the total set of tests available, without differentiating among different propellers and pitch settings. The values of these latter fitting coefficients have been used to make non-dimensional those obtained for the different cases, as in the Table 3.22.

In the following, two PMs are considered. The first one, PM1, makes use of all the values of coefficients a_p and τ derived for each propeller configuration. The second one, PM2, uses only one values resulting from the fitting on the complete dataset, without propeller and pitch setting distinction.

3.5.3 Interpolation Scenario

In the following sections, the performances of the PMs, DDMs, and HMs are tested and compared by means of the data described in two different scenarios, the Interpolation and the Extrapolation one. Basically the two scenarios just differ in the way $\mathcal{D}_n$ and $\mathcal{T}_m$ have been built. In other words, the two scenarios differ in the subset of data exploited for building and testing the models.

For what concerns the PMs, the PM1 and PM2 described in Section 3.5.2 are considered. As described, PM1 and PM2 are just able to predict f_c and RNL_c based on a subset of the input variables described in Table 3.20.

For what regards the DDMs, the custom KRLS algorithm described in Section 2.4.2 is used, adopting the polynomial FM described by Eq. (2.13). DDMs are able to predict f_{bp1}, RNL_{bp1}, f_c, RNL_c, f_{bp2}, RNL_{bp2}, and RNL_b based on all the input variables reported in Table 3.20. The set of hyperparameters tuned during the MS phase are $\mathcal{H} = \{p, c, \lambda\}$ chosen in $\mathfrak{H} = \{1, 2, \cdots, 10\} \times \{10^{-4}, 10^{-3}, \cdots, 10^{+4}\} \times \{10^{-4.0}, 10^{-3.8}, \cdots, 10^{+4.0}\}$.

For the HMs the custom algorithm described in Section 2.4.2 is exploited adopting the polynomial FM described by Eq. (2.13). HMs are able to predict just f_c and RNL_c based on all the input variables reported in Table 3.20 and the PMs. The values of f_{bp1}, RNL_{bp1}, f_{bp2}, RNL_{bp2}, and RNL_b cannot be predicted by HMs, since no PMs are available for these features and, consequently, DDMs have to be used. Finally, since two PMs are available, two HMs, HM1 and HM2, are considered, exploiting, respectively, the PM1 and the PM2. The set of hyperparameters tuned during the MS phase are $\mathcal{H} = \{p, c, \lambda, \theta\}$ chosen in $\mathfrak{H} = \{1, 2, \cdots, 10\} \times \{10^{-4}, 10^{-3}, \cdots, 10^{+4}\} \times \{10^{-4.0}, 10^{-3.8}, \cdots, 10^{+4.0}\} \times \{10^{-4.0}, 10^{-3.8}, \cdots, 10^{+4.0}\}$.

All the tests have been repeated 30 times and the average results are reported, together with their t-student 95% confidence interval, in order to ensure the statistical consistency of the results. The indexes of performance for the frequencies are reported in Hz.

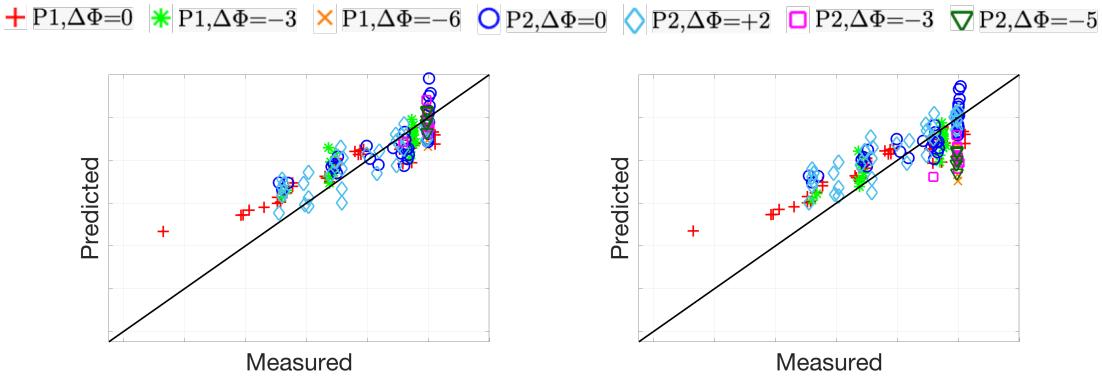
In details, the interpolation case is the simplest one. In this scenario $\mathcal{D}_n$ and $\mathcal{T}_m$ have been created by splitting randomly the whole 164 samples keeping 90% of the data in $\mathcal{D}_n$ and the remaining 10% in $\mathcal{T}_m$. In this way the models have been tested in their ability to predict the propeller noise spectra main characteristics in various, but different, working conditions within the ones exploited for building the model.

The performance of PMs, DDMs, and HMs in predicting the noise spectra main characteristics are reported. The performance are measured with the MAE, the MAPE, and the PPMCC. The table reports the full set of results for completeness. The best performing models are underlined in bold. From the table it is clear how the HMs are in the most cases the best performing ones followed by the DDMs. The PMs are usually the worse performing models. Unfortunately, the full table is not easy to interpret nor very informative and for this reason, in the next sections, a series of scatter plot able to better interpret the results are shown.

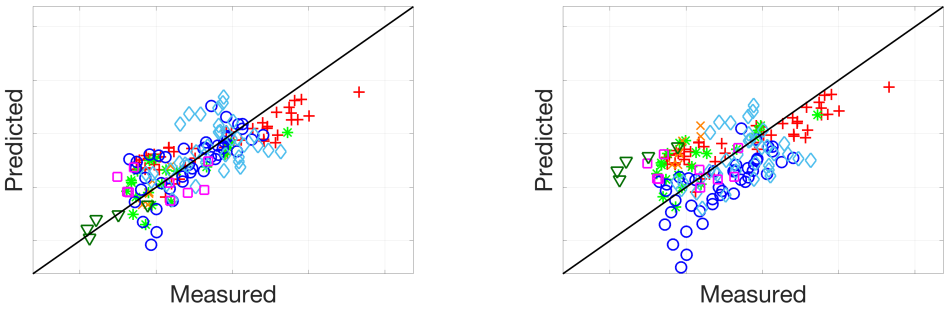
3.5.3.1 Physical Models Results

Figure 3.30 reports the scatter plots of the measured and predicted values of f_c and RNL_c for both PM1 and PM2. From Figure 3.30 it is possible to observe that:

- there is a significant variance of the results depending on the different tests;



(a) f_c, PM1 $[0.2\log_{10}(\text{Hz})/\text{div}]$ (b) f_c, PM2 $[0.2\log_{10}(\text{Hz})/\text{div}]$



(c) RNL_c, PM1 $[10\text{dB}/\text{div}]$ (d) RNL_c, PM2 $[10\text{dB}/\text{div}]$

Figure 3.30: Interpolation Scenario PMs: scatter plots of the measured and predicted values of f_c and RNL_c for both PM1 and PM2.

- especially for f_c, data points tend to distribute with an angle between the measured and predicted values which is slightly different from $45°$, clearly underlining a problem in the PM adopted;
- as expected, the accuracy of the PM1 model is significantly better than the one of the model PM2. However, from a qualitative point of view, results reported in the scatter plots evidence the same problems in both cases.

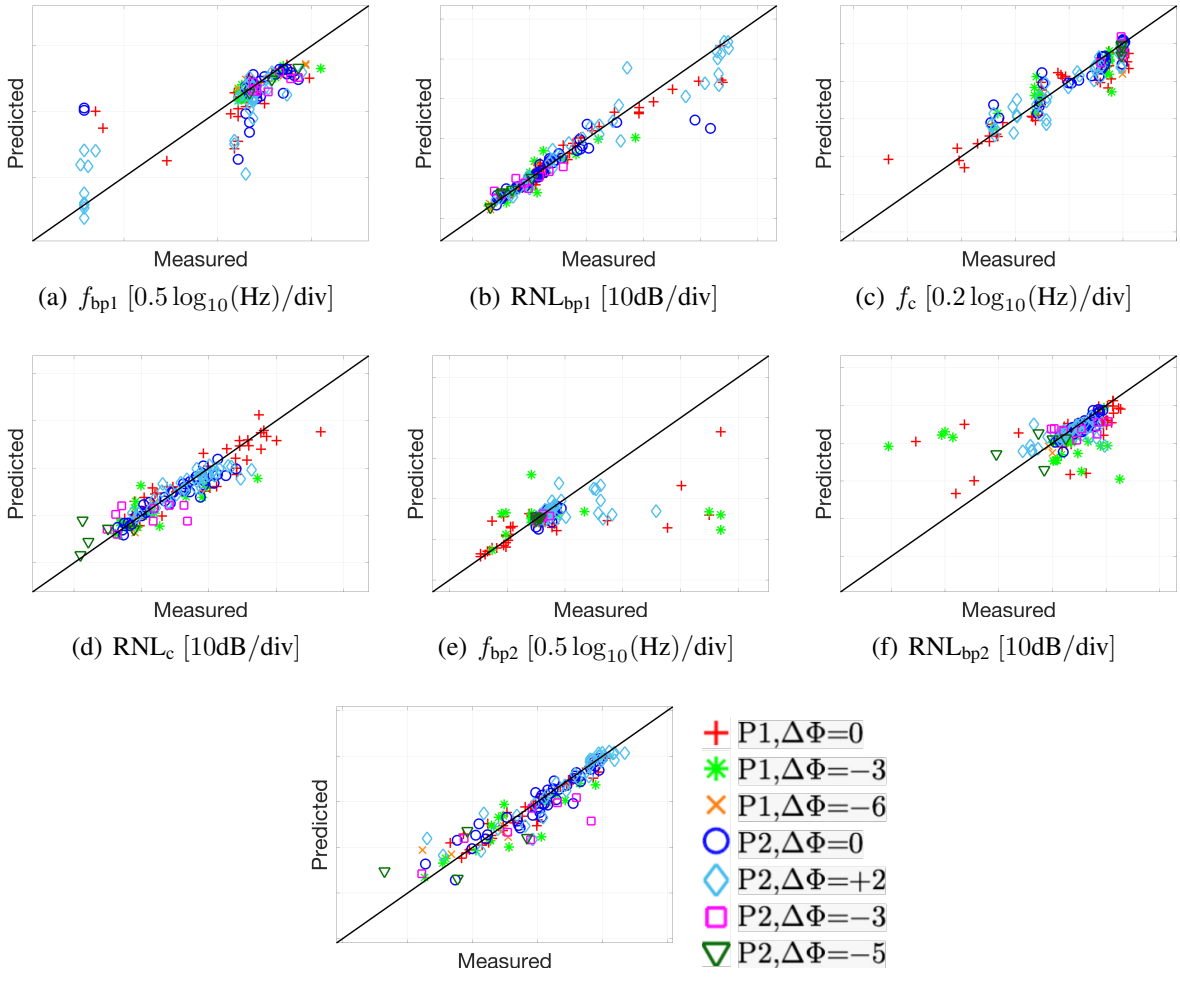
The PM implemented in this work seems to provide only a rough approximation of the dependency of f_c and RNL_c on the input parameters. In the case of the PM1, the values of the parameter τ have been tuned on the available experiments, assuming its value is not dependent on the operational conditions, but only on propeller configuration, while for PM2 τ it is assumed constant on the whole data set. The observed results point out also the limits of these assumptions. Furthermore, one of the main problems related to PMs is that in some cases the noise spectra within the available data present a behaviour similar to the one schematized by the PM whereas in other cases a significantly different behaviour is observed, with the frequency of the maximum weakly depending on cavitation size: the PM is not able to discriminate between these different situations and its accuracy decreases when the cavitation noise is not mainly driven by the tip vortex pulsation, e.g. when the propeller pitch is lower. This may contribute to the significant variance observed.

3.5.3.2 Data-Driven Models Results

Figure 3.31 reports the scatter plots of the measured and predicted values of f_{bp1}, RNL_{bp1}, f_c, RNL_c, f_{bp2}, RNL_{bp2}, and RNL_b for the DDMs. From Figure 3.31 it is possible to observe that:

- considered targets are predicted with a reasonable accuracy by the model, demonstrating that trends present in the experimental data can be effectively modelled by the DDMs;
- the DDMs seems to correctly predict the targets considering also the different behaviours of cavitation noise spectra observed, which represented one of the limits of the considered PMs;
- the results show significantly different trends depending on the considered target
  - a reasonable agreement between measured and predicted data is observed for targets f_c, RNL_c and RNL_b, even if some variance is present;
  - a good agreement is observed also for the target RNL_{bp1}, but in this case there seems to be a deterioration of the prediction performance for the highest values in the distribution;
  - the distributions of points on the scatter diagrams highlight some problems for what regards targets f_{bp1}, f_{bp2} and RNL_{bp2}. Points are partially clustered around certain values, with a significant number of samples spread over the plot without a clear tendency.

These results point out some limits of the definition of the targets for the prediction of noise



(a) f_{bp1} $[0.5\log_{10}(\text{Hz})/\text{div}]$ (b) RNL_{bp1} $[10\text{dB}/\text{div}]$ (c) f_{c} $[0.2\log_{10}(\text{Hz})/\text{div}]$

(d) RNL_{c} $[10\text{dB}/\text{div}]$ (e) f_{bp2} $[0.5\log_{10}(\text{Hz})/\text{div}]$ (f) RNL_{bp2} $[10\text{dB}/\text{div}]$

(g) RNL_{b} $[10\text{dB}/\text{div}]$

Figure 3.31: Interpolation Scenario DDMs: scatter plots of the measured and predicted values of f_{bp1}, RNL_{bp1}, f_{c}, RNL_{fc}, f_{bp2}, RNL_{bp2}, and RNL_{b}.

spectra. The simplified spectral shape adopted effectively succeeds in modelling measured noise spectra, even considering both spectral shapes observed, namely with and without prominent peak.

The issues may be related to the definition of the parameters used to describe the simplified spectra, namely frequencies and levels of the points. Actually, the frequency and levels of the two breakpoints used to define the simplified shape do not feature a clear physical meaning and consequently they are not characterized by clear tendencies as other targets. The first break point, roughly corresponds to the frequency above which the cavitation noise spectrum is perceived over the background noise. As a consequence the frequency of this point, and its corresponding level, depends not only on the characteristics of cavitation noise, but also on the spectrum of the background noise in the tunnel.

The second break point divides the spectrum into two regions characterized by different decay ratio of noise w.r.t. frequency: the decreasing part of the peak, when present, with a larger decay ratio, and the high frequency spectrum with lower decay. These two regions are clearly distinguished only when a prominent peak is identified, otherwise an almost constant decay is observed from the maximum point of the spectrum towards higher frequencies. In the latter case, the definition of the break point is uncertain and consequently f_{bp2} and $\mathrm{RNL}_{\mathrm{bp2}}$ may assume anomalous values. These issues might be overcome in different ways: modifying the simplified representation of spectra or simply employing different parameters to describe it. As an example, the decay ratio (i.e. the slopes of the curve) in the two mentioned regions of the spectrum could be considered in place of the break point. Actually, these parameters should allow to correctly describe the spectra as well, and they should not present anomalous values, even when the distinction between the two regions becomes meaningless. The adoption of alternative parameters and its effect will be subject of future activities.

Evaluating the performance of the DDMs in terms of accuracy is just the first step toward understanding them. In fact, these models are black-box and consequently the learned relation between inputs and outputs is not explicitly known. Since KMs were exploited to extract the explicit form of the model, evaluating the relationship between inputs and outputs would be in most cases impossible while in some cases computational intractable [STC04]. In order to overcome this limitation it was decided to perform a Feature Ranking (FR) procedure [GE03, LM07] which allows to rank the features based on their effect on the model output. For this purpose, a very statistically sound and robust approach called permutation test was exploited [Goo13, FWV06].

Table 3.23 reports the result of the FR procedure on the models which predict f_{c} and $\mathrm{RNL}_{\mathrm{c}}$.

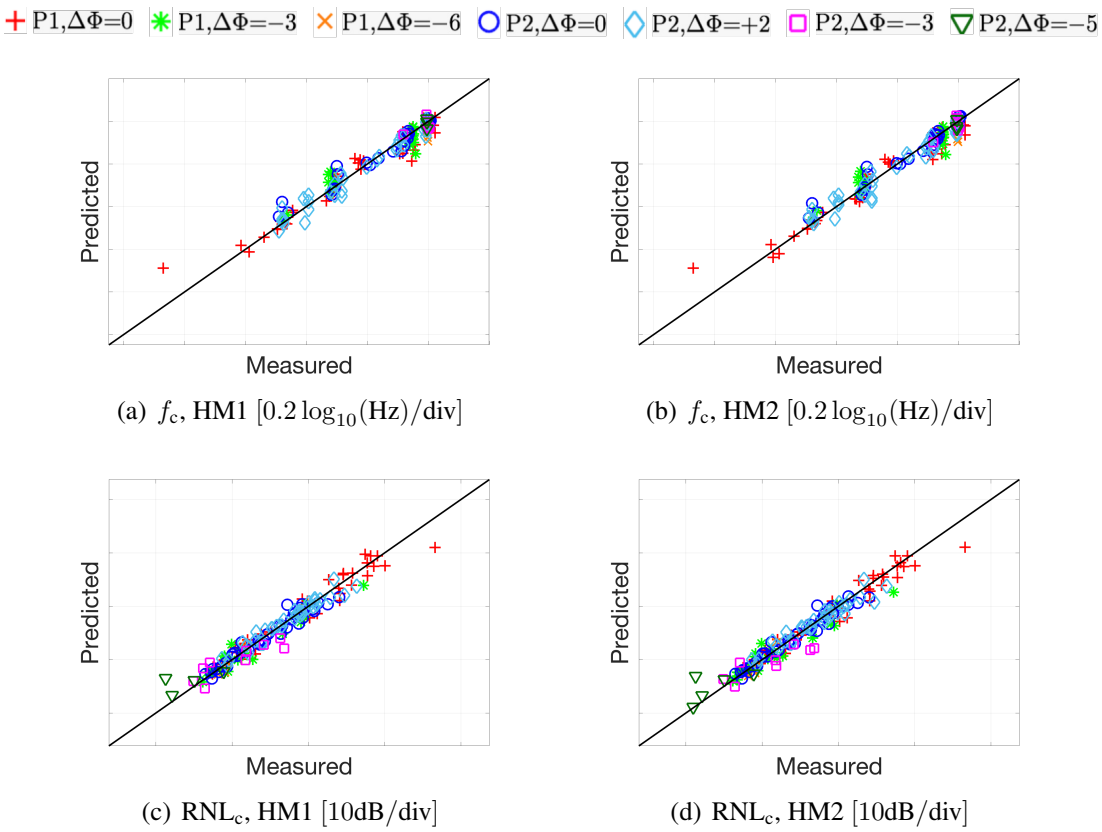
Before analyzing the result of the FR procedure in details, it is worth noting that most of the considered features are strongly dependent or even redundant (e.g. propeller thrust and torque or the alternative definitions of the cavitation number). This means that FR can detect as important just one or all of this strongly correlated features, some of them, or all of them, which basically contain the same information. In fact, one has to remember that the FR procedure is

Table 3.23: Top 20 results of FR on f_c and RNL_c.

f_c		RNL_c	
#	Feature	#	Feature
1	TVC 0°	1	TVC 0°
2	D.TVC	2	S.S. S 0°
3	S.S. S 0°	3	D.TVC
4	max α_{G09}	4	TVC
5	$10K_Q$	5	V_a
6	p_{rel}	6	S.S. RB
7	TVC	7	J
8	n	8	$10K_Q$
9	max α_{G07}	9	P/D
10	Q_c	10	η_o
11	$\overline{\alpha}_{G09}$	11	σ_v
12	σ_v	12	max α_{G07}
13	S.S. S	13	n
14	σ_{ntip}	14	σ_{ntip}
15	σ_n	15	σ_n
16	S.S. RB	16	p_{rel}
17	P/D	17	Q_c
18	min α_{G09}	18	$\Delta\Phi$
19	K_T	19	max α_{G09}
20	T_c	20	T_c

a statistical procedure and consequently subject to uncertainties and statistical fluctuations and, consequently, it does not make sense to make very specific comments while it is more reasonable to observe global trends. Having said this, it is possible observe that:

- not surprisingly, the inception indexes of the driving phenomena (i.e. Tip Vortex at $0°$, Detached Tip Vortex and Suction Side Sheet at $0°$ are always in the Top 3. This confirms the strong relation between measured noise and the cavitation intensity;
- the knowledge summarized in Eq. (3.25) about the resonance frequency of the vortex can be found also in the FR, indeed the torque coefficient (in place of the thrust coefficient), the relative pressure (in place of the cavitation index) and the propeller rotational speed are in the Top 10 features for importance;
- similarly, the sound pressure level of the vortex peak, whose theoretical expression of Eq. (3.26) identifies in the propeller load and the cavitation index the most valuable parameters, is checked by FR;
- in general it can be observed the most influencing variables in the noise generation are among the most important ones according to the FR. Furthermore, it is interesting to notice the absence in the top positions of features assumed to be strongly related to the cavitation



(a) f_c, HM1 $[0.2\log_{10}(\text{Hz})/\text{div}]$ (b) f_c, HM2 $[0.2\log_{10}(\text{Hz})/\text{div}]$

(c) RNL_c, HM1 $[10\text{dB}/\text{div}]$ (d) RNL_c, HM2 $[10\text{dB}/\text{div}]$

Figure 3.32: Interpolation Scenario HMs: scatter plots of the measured and predicted values of f_c and RNL_c for both HM1 and HM2.

noise, as an example the wake parameters. This can be justified recalling that some features, such as the cavitation inception, directly depend on the wake, or some others, as the angle of attack, are derived from it. Hence, wake features could be redundant.

3.5.3.3 Hybrid Models Results

Figure 3.32 reports the scatter plots of the measured and predicted values of f_c and RNL_c for both HM1 and HM2. From Figure 3.32 it is possible to observe that there is a significant enhancing of the performances of the DDMs by using the PMs. Moreover, the results show that the use of a more generic set of coefficients (PM2) does not lead to a noticeable degradation of the performance of the HMs, making them very attractive and promising in view of future enlargements of the experimental dataset.

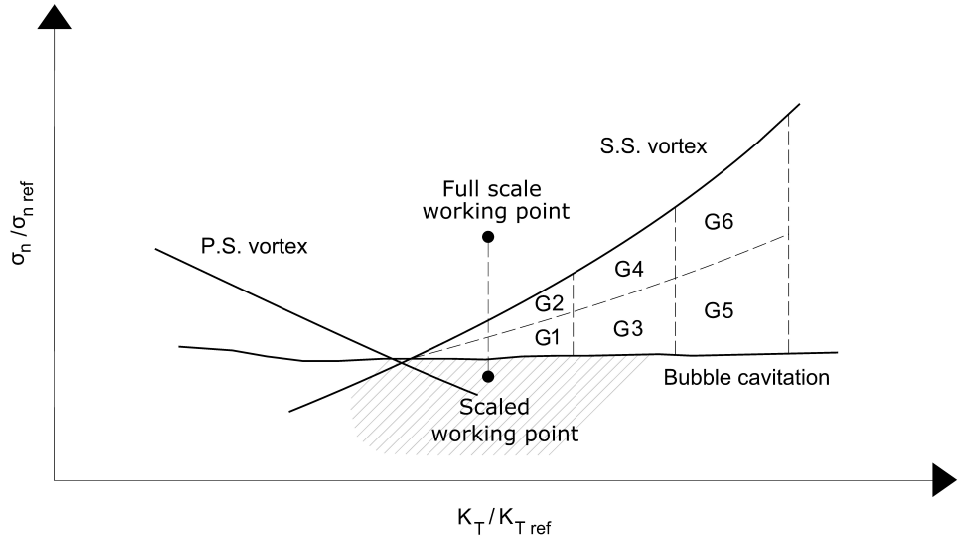


Figure 3.33: Sketch for data domain subdivision for extrapolation tests.

3.5.4 Extrapolation Scenario

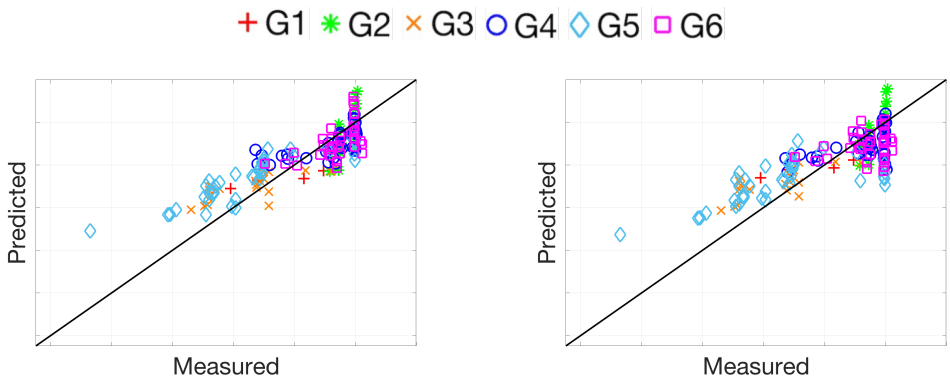
From a practical point of view, it is of great interest to test the capability of the models to predict radiated noise for cases not included in the variable domain of the data used to build them. Actually, new cases of interest might be characterized by values of the input variables not included between those considered, but still similar to them. This may be the case of a new propeller designed with operating requirements different from those considered.

As already anticipated, another important application of the models developed is related to those ship operational conditions for which the cavitation pattern cannot be correctly reproduced in model scale, requiring an extrapolation. In order to better present the problem, the typical cavitation bucket of a model scale propeller is schematized in Figure 3.33. The full scale working point is characterised by the values of the thrust coefficient and cavitation number of the full scale propeller. For this combination of values the model scale propeller does not cavitate, because of viscous scale effects affecting the development of vortices. However, applying scaling formulas like those proposed by [McC62] or [SGJ09], it is possible to assess if TVC is present in full scale and roughly estimate its extension by means of the σ_n/σ_{ninc} ratio or similar quantities. In order to reproduce the same cavitation extension in model scale, some scaling criterion must be applied, such as the identity of the ratio σ_n/σ_{ninc}. The test conditions obtained following this approach (i.e. the scaled working point in the example) are, in some cases, located on the bucket in the hatched area. In this area the presence of unwanted phenomena, like bubble cavitation or other measuring issues (e.g. bubble scattering), may alter the measured noise and also the development of the vortex cavitation itself. As a consequence, the direct measurement in such conditions is deemed not meaningful. Radiated noise in the hatched area should be predicted eliminating the

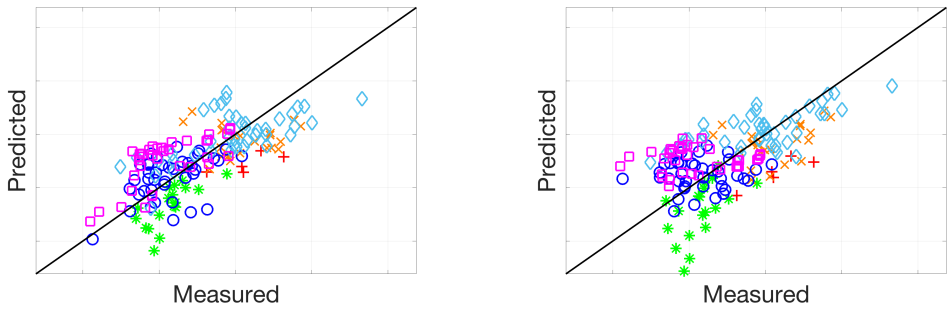
unwanted effects, thus obtaining a sort of ideal model scale experiment. The proposed approach basically consists in developing a model based only on operational conditions without unwanted phenomena. These conditions correspond to the areas identified by numbers from G1 to G6 in Figure 3.33. These areas are defined by combinations of the quantities K_T/K_{Tref} and σ_n/σ_{ninc} of the TVC. The model based on these data is able to predict noise for different extensions of TVC and different combination of input parameters without modelling the unwanted phenomena. The target conditions in the hatched area are characterized by a combination of input parameters not included in the set of data used to define the model, hence an extrapolation is needed. Since validation data for the target region is not available, it is not possible to directly test the capability of the model to extrapolate in this area.

In order to obtain an indication of such capability, the extrapolation performance between different subsets of data, corresponding to the numbered areas in Figure 3.33, has been assessed. In particular the extrapolation test consists in including in $\mathcal{D}_n$ only five of the six groups and use the sixth group as $\mathcal{T}_m$. From the point of view of the physical problem here summarized (i.e. the prediction of the hatched area), the most interesting extrapolation cases are those for which $\mathcal{T}_m$ is represented by points belonging to the zones 1, 2, and 3.

The full set of results are reported, and they are complemented with a series of scatter plots in order to better comment them. Note that, in this case, the results are not checked based on the propeller characteristics but based on the group membership (see Figure 3.33). It is clear that the DDMs and the HMs are able to effectively make predictions in the extrapolation setting and to provide estimation which can be reliable in a real world application of the method showing their ability to extrapolate and not just interpolate. Figures 3.34, 3.35, and 3.36, analogously to Figures 3.30, 3.31, and 3.32 in Section 3.5.3, report the scatter plots of the measured and predicted values of f_{bp1}, RNL_{bp1}, f_{c}, RNL_{c}, f_{bp2}, RNL_{bp2}, and RNL_{b} for the PMs (if available), the DDMs, and the HMs (if available) respectively. The extrapolation performance of the DDMs and HMs are promising, and related results seem to confirm the validity of the proposed approach. The accuracy of the extrapolation is remarkable for all the groups except G5 for which however results are still acceptable. Anyway it has to be remarked that, for the sake of extrapolation to critical conditions previously described, the extrapolation of group G5 is not so important, presenting this group the highest deviations, in terms of cavitating behaviour, from the design conditions. On the contrary, results obtained for the extrapolation of groups G1, G2 and G3 are definitely encouraging.

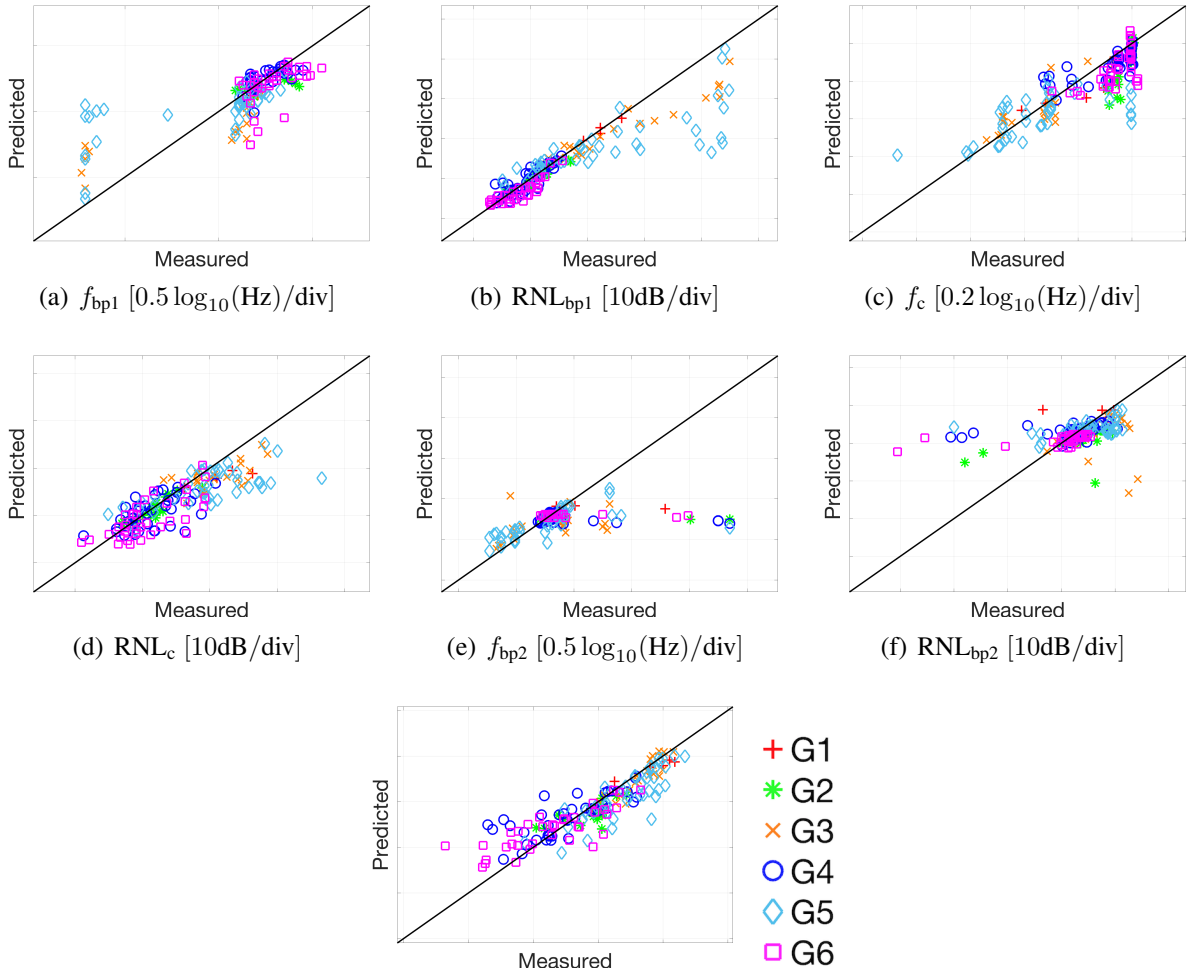


(a) f_c, PM1 $[0.2\log_{10}(\text{Hz})/\text{div}]$ (b) f_c, PM2 $[0.2\log_{10}(\text{Hz})/\text{div}]$



(c) RNL_c, PM1 $[10\text{dB}/\text{div}]$ (d) RNL_c, PM2 $[10\text{dB}/\text{div}]$

Figure 3.34: Extrapolation Scenario PMs: scatter plots of the measured and predicted values of f_c and RNL_c for both PM1 and PM2.

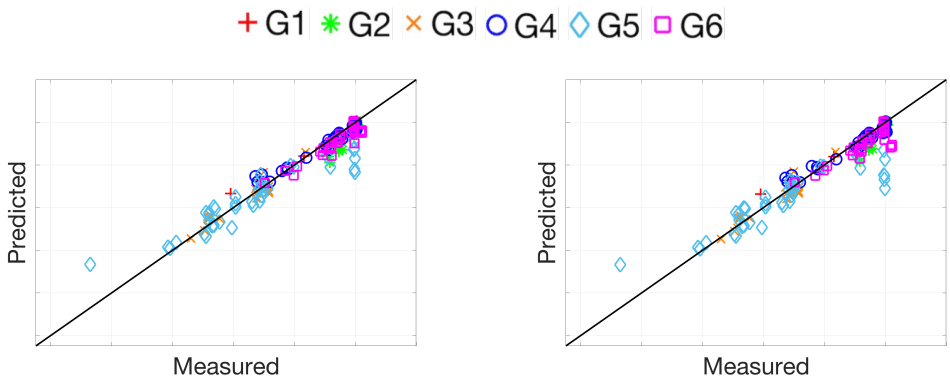


(a) f_{bp1} $[0.5\log_{10}(\mathrm{Hz})/\mathrm{div}]$ (b) $\mathrm{RNL}_{\mathrm{bp1}}$ $[10\mathrm{dB}/\mathrm{div}]$ (c) f_{c} $[0.2\log_{10}(\mathrm{Hz})/\mathrm{div}]$

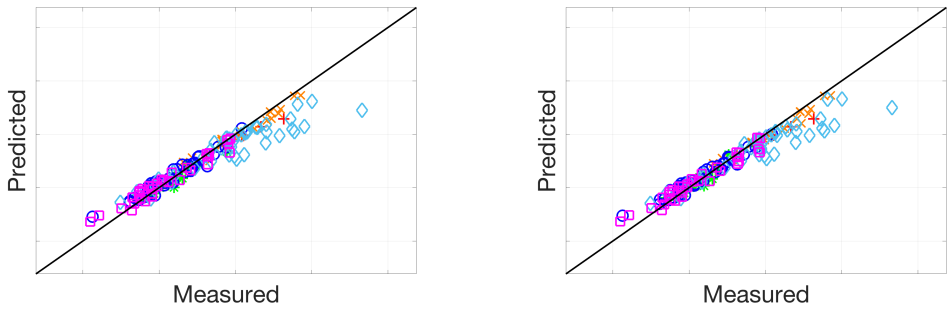
(d) $\mathrm{RNL}_{\mathrm{c}}$ $[10\mathrm{dB}/\mathrm{div}]$ (e) f_{bp2} $[0.5\log_{10}(\mathrm{Hz})/\mathrm{div}]$ (f) $\mathrm{RNL}_{\mathrm{bp2}}$ $[10\mathrm{dB}/\mathrm{div}]$

(g) $\mathrm{RNL}_{\mathrm{b}}$ $[10\mathrm{dB}/\mathrm{div}]$

Figure 3.35: Extrapolation Scenario DDMs: scatter plots of the measured and predicted values of f_{bp1}, $\mathrm{RNL}_{\mathrm{bp1}}$, f_{c}, $\mathrm{RNL}_{\mathrm{c}}$, f_{bp2}, $\mathrm{RNL}_{\mathrm{bp2}}$, and $\mathrm{RNL}_{\mathrm{b}}$.



(a) f_c, HM1 $[0.2\log_{10}(\text{Hz})/\text{div}]$ (b) f_c, HM2 $[0.2\log_{10}(\text{Hz})/\text{div}]$



(c) RNL_c, HM1 $[10\text{dB}/\text{div}]$ (d) RNL_c, HM2 $[10\text{dB}/\text{div}]$

Figure 3.36: Extrapolation Scenario HMs: scatter plots of the measured and predicted values of f_c and RNL_c for both HM1 and HM2.

Chapter 4

Conclusions

Data Analysis is improving our ways to understand complex phenomena by profitably taking advantage of the information behind a collection of data adopting algorithms coming from the world of statistics and Machine Learning. The application of such methods to marine contexts opens new research scenarios, since problems characterised by heterogeneous and numerous data which could not be analysed adopting more classical naval techniques can now be solved with higher accuracy. In fact, the behaviour and interaction of the main components of a ship propulsion systems cannot be easily modelled with a priori physical knowledge, considering the large amount of variables influencing them.

As shown in this thesis, it is possible to build effective data-driven which do not require any a priori knowledge by exploiting the most recent statistical techniques and the large amount of historical data collected by the current on-board automation systems. Moreover, in this thesis, new types of hybrid models are proposed combining empirical and theoretical knowledge in order to take the best part of the two approaches.

The aim of this thesis was to propose data-driven, and where applicable hybrid, solutions to different problems in the naval domain, namely:

- Condition-Based Maintenance;
- Energy and Consumption Monitoring;
- System Safety;
- Components Design.

In order to succeed in this purpose, many models were taken into account, both from the supervised and the unsupervised machine learning domain. From the conducted experiments, some major conclusions came clear.

Condition-Based Maintenance Consisting in assets monitoring, maintenance planning, and real-time anomaly detection, the maintenance of the several components of a Ship Propulsion Systems is an onerous activity, which needs to be efficiently programmed by a shipbuilding company to save time and money. The replacement policies of these components can be planned in a Condition-Based fashion, by predicting their decay state and thus proceed to substitution only when needed. In this context, a prediction of the decay value of each component of the ship can bring a significant limitation of the costs due to maintenance. Moreover, a visualisation of the health condition of the component could provide a compact and expressive representation of their actual state, representing a valuable instrument for decision-making. In this thesis, two main projects concerning Condition-Based Maintenance have been proposed, regarding the comparison of supervised and unsupervised approaches in the prediction of the components decay and a user-friendly visualization of it.

In the first setting, the behavior and interaction of the main components of Ship Propulsion System have been modeled by exploiting the most recent statistical techniques and the large amount of historical data collected by the current on-board automation systems without any a priori knowledge. In particular, the developed models are able to continuously monitor the propulsion equipment to avoid PRM or CM and take decisions based on the actual condition of the propulsion plant. A naval ship, characterized by a COmbined Diesel ELectric And Gas propulsion plant, has been exploited to show the effectiveness of the proposed approaches. From the prediction over many decay components of the ship, it was possible to notice that, as the model grows in complexity, taking into account more than one decay component per time, the error grows. This is caused by the fact that in simpler models, it is clear which one of the decay component is decaying, while in more complex contexts, the prediction model has to take into account the possibility of more than one component decaying per time. Moreover, a dataset containing a detailed information of the decay of the different vessel ship can difficulty be collected in a real-world scenario. In addiction, DMMS require a significant amount of labelled data to efficiently monitor the naval propulsion system components state, which would require the operational stop or even the drydocking of the vessel. Vessels are rarely available in the harbour for maintenance, and the necessary parameters cannot be easily acquired when the ship is operative. In this context, it is possible to take advantage of an indirect measurement of the decay, or by only identifying good and bad working conditions, thus adopting a relaxation on the decay value prediction requirement, by transforming the regression problem into a simpler one by predicting only the necessity of each component to be replaced or not, instead of the precise component decay.

In the second setting, the problem of assessing and visualizing the health condition of some Induction Motor bearings was inspected. To detect faults in the bearings, contrarily to the state-of-the-art approaches exploiting vibration signals, collected by easily damageable and intrusive vibration probes, the stator currents signals are exploited, which are already commonly available, or easily and unintrusively collectable. Moreover, it was showed that by using state-of-the-art deep neural networks, instead of the now classic techniques like the PCA, it is possible to extract

from the stator current signal a compact and expressive representation of the bearings state, ultimately providing a bearing fault detection visualisation system. In fact, by proposing an image in a low dimensional space representing the bearings state that can be interpreted by an operator, it is possible to give a meaningful representation of the IMs bearing fault phenomena. By exploiting a series of real-data collected from an inverter-fed motor mounting different artificially damaged bearings, the effectiveness of their proposal was showed.

Energy and Consumption Monitoring In order to reduce vessel fuel consumption and gas emissions, the deployment of new solutions has seen a growing interests in the latest years, mainly due the environmental issues the world is facing.

In particular, the problem of estimating the speed loss of the ship has been extensively addressed, since understanding when the hull and propeller loss in performance bring to an excessive fuel consumption represent a valuable information for the ship manufacturers. Since marine fouling is a phenomenon that strongly affects a ship's regarding powering performance and its effects can be observed after just a few months of operations, the possibility of correctly estimate its impacts can improve the ability of the ship operators to effectively schedule the dry-docking for cleaning the hull and the propeller. Moreover, since the predicted speed loss is subject to local noises, mainly due to the weather and operational conditions, the global speed loss trend has only to be seen on a wide time-span in order to understand its behaviour. For this purpose a two-step data-driven approach is proposed, based on the Deep Extreme Learning Machines which are among the most advanced tools in the context of advanced statistical methods. Thanks to such an approach, it is possible to build a Digital Twin of the ship that can be effectively exploited to detect during real operations a deviation in the speed performances (with respect to the ones achievable with clean hull and propeller), and consequently to identify the extension of the marine fouling phenomena. Then the proposal has been compared with the state-of-the-art alternative method, namely the ISO 19030 standard, using real-world data coming from two Handymax chemical/product tankers. Results clearly show the effectiveness of the proposal and its better prediction accuracy and reliability, with respect to the ISO 19030. This is shown by both a more accurate and consistent prediction of the loss of performance over time, between cleaning intervals, and by the ability of automatically detecting maintenance events.

In the future, the proposed method could be exploited also for the evaluation of the effectiveness of different energy-saving solutions, such as the case of a new propeller design or the evaluation of the benefits deriving from the application of sails. The contribution of this work can be seen as a step forward in supporting both the development of new technologies, able to improve performances and efficiency of the ship, and the implementation of suitable Condition-Based Maintenance policies for increasing the shipping sustainability. Moreover, the proposed method, will facilitate the verification of the impact of new technologies or vessel components, thereby allowing to increase the transparency of energy and fuels efficiency technologies by providing a method to validate fuel savings claims made by the manufacturers and providers, supporting

further uptake in the shipping industry.

System Safety Vessels security during maneuvering and collision avoidance was proved to be a significant issue in the latest years. Being able to estimate the vessel safety properties during the preliminary design stages by predicting a vessel crash stop maneuvering performance is a step forward in the direction of a smart and safe ship design since it allows bettering forecast the safety properties of a ship before its production.

In this work, a series of data-driven models able to estimate the vessel's safe manoeuvring properties during the preliminary design stages were developed. This work can be considered a step forward in the direction of a smart and safe ship design and operation. The method presented allows welling forecast the safety-related behaviour of a ship. This can be used to assess the effectiveness of the design before production as well as the suitability of an existing ship for operating in certain restricted areas. Moreover, the proposed data-driven crash stop prediction model can be combined with collision avoidance algorithms with particular focus on the unmanned surface vehicles and autonomous ships. In particular, a crash stop performance prediction which can accurately forecast the results of this safety test was proposed. To achieve this goal, a recent improvement of the RF learning algorithm is reported and show that an accurate tuning procedure can remarkably improve their predictive power. Results on real-world data, collected and provided by DAMEN Shipyards, demonstrate the effectiveness of the proposal.

Components Design Have a direct and simple simulation of a component is important for detecting possible defects in advance; in fact, being able to predict the main characteristics and defects of a vessel component at design stage, only being given some easily-collectable geometry information, will help the design of new components.

In particular, in this thesis, a procedure to estimate propeller cavitation noise by means of numerical models has been presented. The goal is to predict the significant characteristics of the cavitation noise spectra using data available at propeller design stage. This will help the designer to implement suitable countermeasures to mitigate the generation of unwanted noise, limiting one of the major sources of anthropogenic noise. With this purpose in mind, a simplification of the propeller noise spectra has been proposed and a model able to predict its features on the base of quantities that can be estimated at design stage has been developed. Three different modelization strategies have been presented: one based on the physical knowledge of the problem, one based on data science and one based on a hybrid approach able to exploit both the two sources of information. An extensive set of cavitation tunnel tests performed with different propellers and different configurations allowed the creation of a database exploited for developing and testing the different models. An in depth comparison of the performance of the different models has been performed, leading to the conclusion that the hybrid alternative have shown remarkable and promising results, thus opening the way to future works in this direction. The presented analysis

has been limited only to a class of similar propellers and configurations deemed of remarkable interest, in order to generate a collection of data with a tolerable variance. It is planned for future activities to enlarge the data collection considering further propellers and wake fields. This will require the use of an enlarged set of features, including more parameters describing propeller geometries (e.g. expanded area ratio, camber, chord, etc.).

Closing Remarks In conclusion, this study proved that it is possible to treat some major naval problems in a data-driven, or hybrid, fashion adopting machine learning techniques. It has to be noted that the experiments carried out are in many cases relative to one or two specific vessels, so this work cannot be straightforwardly mapped to any new ship without a further investigation on transfer learning. It has to be taken into account that datasets in this field are going to grow in the future, since the sensors implant on-board is going to grow and it is natural to think that the data collected are going to be exploited for monitoring vessels performances. Moreover, data from every fleet vessel could be collected and there would be the possibility to generalize the results obtained through this thesis to a broader spectrum of ships.

In the future, these technologies will be determinant for business strategies, but larger amount of data will require parallel and scalable technologies, which were only rapidly touched during this thesis. As many methods applied in this work are parallelizable, it becomes natural to think of future expansions of these algorithms by developing and programing through distributed technologies. They would allow studying great amount of data as soon as they arrive to a datacenter, through their on streaming applications. This great saving of time could be determinant in case of malfunctioning of a ship and necessity to communicate to the crew on-board the proper behavior to overtake.

Appendix A

Related Publications and Research Activities

The proposed thesis is the result of a three years works in the DA field, developing Machine Learning algorithm for the presented applications. During these years, the following articles have been presented to the international research community, involving the topics covered in this thesis:

- F. Cipollini, A. Coraddu, and L. Oneto, A deep learning approach to marine propulsion system maintenance, 3rd International Symposium on Naval Architecture and Maritime (INT-NAM), Pag: 687-698, 2018;
- F. Cipollini, L. Oneto, A. Coraddu, A. J. Murphy, and D. Anguita, Condition-Based maintenance of naval propulsion systems: Data analysis with minimal feedback, Reliability Engineering & System Safety, 177:12–23, 2018;
- F. Cipollini, L. Oneto, A. Coraddu, A. J. Murphy, and D. Anguita. Condition-Based maintenance of naval propulsion systems with supervised data analysis, Ocean Engineering, 149:268–278, 2018;
- F. Cipollini, L. Oneto, A. Coraddu, and S. Savio, Unsupervised deep learning for induction motor bearings monitoring, Data-Enabled Discovery and Applications, 3(1):1, 2019;
- F. Cipollini, L. Oneto, A. Coraddu, S. Savio, and D. Anguita, Unintrusive monitoring of induction motors bearings via deep learning on stator currents, volume 144, pages 42–51, 2018;
- Oneto L., Baldi F., Cipollini F., Coraddu, A., Estimating marine fouling speedloss with iso 19030: Pros and cons of the standard, SNAMES 38th journal, 2019;
- A. Coraddu, L. Oneto, F. Baldi, F. Cipollini, M. Atlar, and S. Savio, Data-driven ship digital twin for estimating the speed loss due to the marine fouling, Ocean Engineering, 186:106063, 2019;
- L. Oneto, A. Coraddu, P. Sanetti, O. Karpenko, F. Cipollini, T. Cleophas, and D. Anguita,

Marine safety and data analytics: Vessel crash stop maneuvering performance prediction, In International conference on artificial neural networks, pages 385–393, 2017;

- L. Oneto, A. Coraddu, F. Cipollini, O. Karpenko, Sanetti P. Xepapa, K., and D. Anguita, Crash stop maneuvering performance prediction: a data driven solution for safety and collision avoidance, Data-Enabled Discovery and Applications, 2(1):11, 2018;
- F. Cipollini, F. Miglianti, L. Oneto, G. Tani, and M. Viviani, Cavitation noise spectra prediction with hybrid models, INNS Big Data and Deep Learning conference, Pag: 152–157, 2019;
- F. Cipollini, F. Miglianti, L. Oneto, G. Tani, and M. Viviani, Hybrid model for cavitation noise spectra prediction, In IEEE International Joint Conference on Neural Networks (IJCNN), 2019;
- F. Miglianti, F. Cipollini, L. Oneto, G. Tani, and M. Viviani, Model scale cavitation noise spectra modelling: Combining physical knowledge with data science, Ocean Engineering, 2019;
- Miglianti, F., Tani G., Viviani M., Cipollini F., Oneto L., Data-Driven Models for propeller cavitation noise in model scale, In Sixth International Conference on Marine Propulsors (SMP19), Rome, May 27-30, 2019;
- L. Oneto, F. Cipollini, S. Ridella, and D. Anguita, Randomized learning: Generalization performance of old and new theoretically grounded algorithms, Neurocomputing, 298:21–33, 2018;
- A. Carrega, F. Cipollini, and L. Oneto, Simple continuous optimal regions of the space of data, Neurocomputing, 349:91–104, 2019.

Many of these publications have been presented during International conferences:

- IEEE International Conference on Big Data 2017 (IEEE Big Data 2017), 11 – 14 December 2017, at Boston, MA, USA presenting the paper Crack Random Forest for Arbitrary Large Datasets, by A. Lulli, L. Oneto, and D. Anguita;
- INNS Conference on Big Data and Deep Learning (INNS BDDL), April 17 – 19, 2018 at Sanur, Bali, Indonesia, presenting the paper Unintrusive Monitoring of Induction Motors Bearings via Deep Learning on Stator Currents, by F. Cipollini, L. Oneto, A. Coraddu, S. Savio, D. Anguita;
- INNS BDDL 2019, Sestri Levante 16-18 April conference, presenting "Cavitation Noise Spectra Prediction with Hybrid Models";
- IJCNN 2019, Budapest 14-19 July conference, presenting "Hybrid Model for Cavitation Noise Spectra Prediction" and "Ensemble Application of Transfer Learning and Sample Weighting for Stock Market Prediction".

These works have been developed in collaboration with other academic institutions and companies, namely:

- Naval Architecture, Ocean & Marine Engineering, Strathclyde University, Glasgow G1 1XW, UK;

- Marine, Offshore and Subsea Technology Group, School of Engineering, Newcastle University, Newcastle upon Tyne, NE1 7RU, UK;
- DITEN - University of Genova, Via Opera Pia 11A, I-16145 Genova, Italy;
- Ecole Polytechnique Fédérale de Lausanne, EnergyPolis, 1950 Sione;
- Research & Technology Support - Damen Schelde Naval Shipbuilding, Netherlands;
- AizoOn Technology Consulting, Torre San Vincenzo, Via San Vincenzo 2, 16-th floor, I-16121 Genova, Italy;

At the same time, I had the opportunity to follow different academic courses, which enabled me to develop the knowledge and ability to become a Data Analysist:

- Scientific Writing;
- Machine Learning Crash Course;
- Industrial ML: Theory and Practice of Learning from Data;
- Introduction to Convex Optimization;
- Programming Complex Heterogeneous Parallel Systems;
- Languages and Solving Techniques for Knowledge Representation and Reasoning;
- Adding Value to Data, Extending Database Technology Summer School;
- Regularization Methods for Machine Learning;
- Large-Scale Data Management;
- Deep Learn 2018 Summer School.

Bibliography

[AAF08] T. S. Akinfiev, M. A. Armada, and R. Fernandez. Nondestructive testing of the state of a ship's hull with an underwater robot. *Russian Journal of Nondestructive Testing*, 44(9):626–633, 2008.

[AAFS18] B. Aktas, M. Atlar, P. Fitzsimmons, and W. Shi. An advanced joint time-frequency analysis procedure to study cavitation-induced noise by using standard series propeller data. *Ocean Engineering*, 170:329–350, 2018.

[AB12] K. Anderson and A. Bows. Executing a Scharnow turn: reconciling shipping emissions with international commitments on climate change. *Carbon Management*, 3(6):615–628, 2012.

[ABFC09] M. Altosole, G. Benvenuto, M. Figari, and U. Campora. Real-time simulation of a cogag naval ship propulsion system. *Journal of Engineering for the Maritime Environment*, 223(1):47–62, 2009.

[ABLWB16] K. Andersson, S. Brynolf, F. Lindgren, and M. Wilewska-Bien. *Shipping and the environment: improving environmental performance in marine transportation*. Springer, 2016.

[ABR00] D. Anguita, A. Boni, and S. Ridella. Evaluating the generalization ability of support vector machines through the bootstrap. *Neural Processing Letters*, 11(1):51–58, 2000.

[AC10] S. Arlot and A. Celisse. A survey of cross-validation procedures for model selection. *Statistics surveys*, 4:40–79, 2010.

[ACJW18] R. Adland, P. Cariou, H. Jia, and F.C. Wolff. The energy efficiency effects of periodic ship hull cleaning. *Journal of Cleaner Production*, 178:1–13, 2018.

[ACS05] B. Ayhan, M.-Y. Chow, and M.-H. Song. Multiple signature processing-based fault detection schemes for broken rotor bar in induction motors. *IEEE Transactions on Energy Conversion*, 20:336–343, 2005.

[ACS06] B. Ayhan, M.-Y. Chow, and M.-H. Song. Multiple discriminant analysis and neural-network-based monolith and partition fault-detection schemes for broken rotor bar in induction motors. *IEEE Transactions on Industrial Electronics*, 53:1298–1308, 2006.

[AEP08] A. Argyriou, T. Evgeniou, and M. Pontil. Convex multi-task feature learning. *Machine Learning*, 73(3):243–272, 2008.

[AES10] A. E. B. Abu-Elanien and M. M. A. Salama. Asset management techniques for transformers. *Electric power systems research*, 80(4):456–464, 2010.

[AGC+02] M. Atlar, E. J. Glover, M. Candries, R. J. Mutton, and C. D. Anderson. The effect of a foul release coating on propeller performance. In *International conference on Marine Science and Technology for Environmental Sustainability (ENSUS 2002)*, 2002.

[AGO+13] D. Anguita, A. Ghio, L. Oneto, X. Parra, and J. L. Reyes-Ortiz. A public domain dataset for human activity recognition using smartphones. In *ESANN*, 2013.

[AGOR12] D. Anguita, A. Ghio, L. Oneto, and S. Ridella. In-sample and out-of-sample model selection and error estimation for support vector machines. *IEEE Transactions on Neural Networks and Learning Systems*, 23:1390–1406, 2012.

[Ame06] American Bureau of Shipping. Guide for vessel manoeuvrability, 2006.

[AOTR08] B. Akin, U. Orguner, H. A. Toliyat, and M. Rayner. Low order pwm inverter harmonics contributions to the inverter-fed induction machine fault diagnosis. *IEEE Transactions on Industrial Electronics*, 55:610–619, 2008.

[AYT+18] M. Atlar, I. Yeginbayeva, S. Turkmen, Y. Demirel, S. Carchen, A. Marino, and D. Williams. A rational approach to predicting the effect of fouling control systems on "in-service" ship performance. In *3rd International Conference on Naval Architecture and Maritime, Yildiz Technical University, Istanbul (INTNAM 2018)*, 2018.

[Bal06] R. Balkin. The international maritime organization and maritime security. *Tul. Mar. LJ*, 30:1, 2006.

[Bax00] J. Baxter. A model of inductive bias learning. *Journal of artificial intelligence research*, 12:149–198, 2000.

[BB09] G. Budai-Balke. *Operations research models for scheduling railway infrastructure maintenance*. Rozenberg Publishers, 2009.

[BB10a] R.E. Bensow and G. Bark. Implicit les predictions of the cavitating flow on a propeller. *Journal of Fluids Engineering*, 132:41302, 2010.

[BB10b] R.E. Bensow and G. Bark. Simulating cavitating flows with les in openfoam. In *Proceedings of the 5th European Conference on Computational Fluid Dynamics*, 2010.

[BB12] J. Bergstra and Y. Bengio. Random search for hyper-parameter optimization. *Journal of Machine Learning Research*, 13:281–305, 2012.

[BCV13] Y. Bengio, A. Courville, and P. Vincent. Representation learning: A review and new perspectives. *IEEE transactions on pattern analysis and machine intelligence*, 35:1798–1828, 2013.

[Ben00] M. E. H. Benbouzid. A review of induction motors signature analysis as a medium for faults detection. *IEEE transactions on industrial electronics*, 47:984–993, 2000.

[Ben09] Y. Bengio. Learning deep architectures for ai. *Foundations and trends® in Machine Learning*, 2:1–127, 2009.

[BF15] R. Blaser and P. Fryzlewicz. Random rotation ensembles. *JMLR*, 2:1–15, 2015.

[BFF13] L. Briançon, P. Fournier, and D. Fréchou. Marine propeller noise measurements techniques in hydroacoustics tunnel. In *The 3rd International Conference on Advanced Model Measurement Technology for EU Maritime Industry, Gdansk, Poland*, 2013.

[BGRR08] M. Blodt, P. Granjon, B. Raison, and G. Rostaing. Models for bearing damage detection in induction motors using stator current monitoring. *IEEE transactions on industrial electronics*, 55:1813–1822, 2008.

[BH03] B. Bakker and T. Heskes. Task clustering and gating for bayesian multitask learning. *Journal of Machine Learning Research*, 4:83–99, 2003.

[Bis95] C. M. Bishop. *Neural networks for pattern recognition*. Oxford university press, 1995.

[BK88] H. Bourlard and Y. Kamp. Auto-association by multilayer perceptrons and singular value decomposition. *Biological cybernetics*, 59:291–294, 1988.

[BK03] M. E. H. Benbouzid and G. B. Kliman. What stator current processing-based technique to use for induction motor rotor faults diagnosis? *IEEE Transactions on Energy Conversion*, 18:238–244, 2003.

[BK16] N. Bialystocki and D. Konovessis. On the estimation of ship's fuel consumption and speed curve: A statistical approach. *Journal of Ocean Engineering and Science*, 1(2):157–166, 2016.

[BL12] M. A. Beyer and D. Laney. The importance of 'big data': a definition. *Stamford, CT: Gartner*, pages 2014–2018, 2012.

[BLS+13] S. Bagavathiappan, B. B. Lahiri, T. Saravanan, J. Philip, and T. Jayakumar. Infrared thermography for condition monitoring-a review. *Infrared Physics & Technology*, 60:35–55, 2013.

[Bos09] J. Bosschers. Investigation of hull pressure fluctuations generated by cavitating vortices. In *Proc. First Symposium on Marine Propulsors*, 2009.

[Bos18] J. Bosschers. A semi-empirical prediction method for broadband hull-pressure fluctuations and underwater radiated noise by propeller tip vortex cavitation. *Journal of Marine Science and Engineering*, 6(2):49, 2018.

[BR16] S. Boschert and R. Rosen. Digital twin—the simulation aspect. In *Mechatronic Futures*, 2016.

[Bre01] L. Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.

[BU15] O. C. Basurko and Z. Uriondo. Condition-based maintenance for medium speed diesel engines used in vessels in operation. *Applied Thermal Engineering*, 80:404–412, 2015.

[CAA03] M. Candries, M. Atlar, and C. D. Anderson. Estimating the impact of new-generation antifoulings on ship performance: the presence of slime. *Journal of Marine Engineering & Technology*, 2(1):13–22, 2003.

[Cal92] N. Calder. *Marine Diesel engines: Maintenance, troubleshooting, and repair*. International Marine, 1992.

[Car97] R. Caruana. Multitask learning. *Machine Learning*, 28(1):41–75, 1997.

[Car12] J. Carlton. *Marine propellers and propulsion*. Butterworth-Heinemann, 2012.

[Cat07] O. Catoni. *PAC-Bayesian Supervised Classification*. Institute of Mathematical Statistics Lecture Notes-Monograph Series, 2007.

[CDV07] A. Caponnetto and E. De Vito. Optimal rates for the regularized least-squares algorithm. *Foundations of Computational Mathematics*, 7(3):331–368, 2007.

[CH67] T. Cover and P. Hart. Nearest neighbor pattern classification. *IEEE transactions on information theory*, 13(1):21–27, 1967.

[CH13] E. Cambria and G.-B. Huang. Extreme learning machines. *IEEE Intelligent Systems*, 28(6):30–59, 2013.

[Cha63] W. Chauvenet. *A Manual of Spherical and Practical Astronomy*. Lippincott, 1863.

[COBA17] A. Coraddu, L. Oneto, F. Baldi, and D. Anguita. Vessels fuel consumption forecast and trim optimisation: A data analytics perspective. *Ocean Engineering*, 130:351–370, 2017.

[COB+ed] A. Coraddu, L. Oneto, F. Baldi, F. Cipollini, S. Savio, and D. Anguita. Data-driven ship digital twin for estimating the speed loss due to the marine fouling. *Ocean Engineering*, submitted.

[COC+18a] F. Cipollini, L. Oneto, A. Coraddu, A. J. Murphy, and D. Anguita. Condition-based maintenance of naval propulsion systems: Data analysis with minimal feedback. *Reliability Engineering & System Safety*, 177:12–23, 2018.

[COC+18b] F. Cipollini, L. Oneto, A. Coraddu, A. J. Murphy, and D. Anguita. Condition-based maintenance of naval propulsion systems with supervised data analysis. *Ocean Engineering*, 149:268–278, 2018.

[COC+18c] F. Cipollini, L. Oneto, A. Coraddu, S. Savio, and D. Anguita. Unintrusive monitoring of induction motors bearings via deep learning on stator currents. In *Procedia computer science*, 2018.

[COG+16] A. Coraddu, L. Oneto, A. Ghio, S. Savio, D. Anguita, and M. Figari. Machine learning approaches for improving condition-based maintenance of naval propulsion plants. *Proceedings of the Institution of Mechanical Engineers, Part M: Journal of Engineering for the Maritime Environment*, 230:136–153, 2016.

[CPA17] A. Carchen, K. Pazouki, and M. Atlar. A rational approach to predicting the effect of fouling control systems on "in-service" ship performance. In *Hull Performance and Insight Conference (HullPIC)*, 2017.

[Cra93] S. Crainer. *Zeebrugge: Learning from Disaster: Lessons in Corporate Responsibility*. Herald Charitable Trust, 1993.

[CSC11] CSC. Air pollution and energy efficiency, a transparent and reliable hull and propeller performance standard. Technical report, Clean Shipping Coalition, 2011.

[CST00] N. Cristianini and J. Shawe-Taylor. *An introduction to support vector machines and other kernel-based learning methods*. Cambridge university press, 2000.

[DAK15] N. Doerry, J. Amy, and C. Krolick. History and the status of electric ship propulsion, integrated power systems, and future trends in the us navy. *Proceedings of the IEEE*, 103(12):2243–2251, 2015.

[Dek96] R. Dekker. Applications of maintenance optimization models: a review and analysis. *Reliability engineering & system safety*, 51:229–240, 1996.

[DGDLM13] Y. Demchenko, P. Grosso, C. De Laat, and P. Membrey. Addressing big data issues in scientific data infrastructure. In *2013 International Conference on Collaboration Technologies and Systems (CTS)*, 2013.

[Dir08] Directive, Strategy Framework. Directive 2008/56/EC of the European Parliament and of the Council. *Official Journal of the European Union*, 2008.

[DKTA13] P. C. Deshpande, P. P. Kalbar, A. K. Tilwankar, and S. R. Asolekar. A novel approach to estimating resource consumption rates and emission factors for ship recycling yards in alang, india. *Journal of cleaner production*, 59:251–259, 2013.

[DRK14] H. Dankowski, P. Russell, and S. Krüger. New insights into the flooding sequence of the costa concordia accident. In *ASME 2014 33rd international conference on ocean, offshore and arctic engineering*, 2014.

[DTI17] Y. K. Demirel, O. Turan, and A. Incecik. Predicting the effect of biofouling on ship resistance using cfd. *Applied Ocean Research*, 62:100 – 118, 2017.

[Duc00] W. Duch. Similarity-based methods: a general framework for classification, approximation and association. *Control and Cybernetics*, 29(4), 2000.

[DUZ+17] Y. K. Demirel, D. Uzun, Y. Zhang, H. Fang, A. H. Day, and O. Turan. Effect of barnacle fouling on ship resistance and powering. *Biofouling*, 33(10):819–834, 2017.

[EP04] T. Evgeniou and M. Pontil. Regularized multi–task learning. In *ACM SIGKDD international conference on Knowledge discovery and data mining*, 2004.

[ET94] B. Efron and R. J. Tibshirani. *An introduction to the bootstrap*. CRC press, 1994.

[FB10] L. Frosini and E. Bassi. Stator current and motor efficiency as indicators for different types of bearing faults in induction motors. *IEEE Transactions on Industrial electronics*, 57:244–251, 2010.

[FFTV00] F. Filippetti, G. Franceschini, C. Tassoni, and P. Vas. Recent developments of induction motor drives fault diagnosis using ai techniques. *IEEE transactions on industrial electronics*, 47:994–1004, 2000.

[FTK17] M.I. Foteinos, E.I. Tzanos, and N.P. Kyrtatos. Ship hull fouling estimation using shipboard measurements, models for resistance components, and shaft torque calculation using engine model. *Journal of Ship Research*, 61(2):64–74, 2017.

[FUI06] T. Fujiwara, M. Ueno, and Y. Ikeda. Cruising Performance of a Large Passenger Ship In Heavy Sea. In *The Sixteenth International Offshore and Polar Engineering Conference*, 2006.

[FWV06] D. François, V. Wertz, and M. Verleysen. The permutation test for feature selection by mutual information. In *European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning*, 2006.

[GCSR14] A. Gelman, J. B. Carlin, H. S. Stern, and D. B. Rubin. *Bayesian data analysis*. Chapman and Hall/CRC, 2014.

[GE03] I. Guyon and A. Elisseeff. An introduction to variable and feature selection. *Journal of machine learning research*, 3:1157–1182, 2003.

[GGA13] L. Ghelardoni, A. Ghio, and D. Anguita. Energy load forecasting using empirical mode decomposition and support vector regression. *IEEE Transactions on Smart Grid*, 4(1):549–556, 2013.

[GGHGGM+05] A. García-Gamboa, N. Hernández-Gress, M. González-Mendoza, R. Ibarra-Orozco, and J. Mora-Vargas. A comparison of different initialization strategies to reduce the training time of support vector machines. In *International Conference on Artificial Neural Networks*, 2005.

[Giu05] P. Giudici. *Applied data mining: statistical methods for business and industry*. John Wiley & Sons, 2005.

[Gla71] J. W. L. Glaisher. On a class of definite integrals. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 42(280):294–302, 1871.

[GLLF15] P. Germain, A. Lacasse, M. Laviolette, A. ahd Marchand, and Roy J. F. Risk bounds for the majority vote: From a pac-bayesian analysis to a learning algorithm. *JMLR*, 16(4):787–860, 2015.

[GLLP04] N. Gebraeel, M. Lawley, R. Liu, and V. Parmeshwaran. Residual life predictions from vibration-based degradation signals: a neural network approach. *IEEE Transactions on industrial electronics*, 51:694–700, 2004.

[Goo13] P. Good. *Permutation tests: a practical guide to resampling methods for testing hypotheses*. Springer Science & Business Media, 2013.

[GRRE12] H. M. Gaspar, A. M. Ross, D. H. Rhodes, and S. O. Erikstad. Handling complexity aspects in conceptual ship design. In *International Maritime Design Conference, Glasgow, UK*, 2012.

[GS12] E. Glaessgen and D. Stargel. The digital twin paradigm for future nasa and us air force vehicles. In *53rd Structures, Structural Dynamics and Materials Conference*, 2012.

[GTVC14] S. Gaggero, G. Tani, M. Viviani, and F. Conti. A study on the numerical prediction of propellers cavitating tip vortex. *Ocean engineering*, 92:137–161, 2014.

[GWT+18] P. Gilbert, C. Walsh, M. Traut, U. Kesieme, K. Pazouki, and A. Murphy. Assessment of full life-cycle air emissions of alternative shipping fuels. *Journal of Cleaner Production*, 172:855–866, 2018.

[GYO10] A. Ghasemi, S. Yacout, and M. S. Ouali. Parameter estimation methods for condition-based maintenance with indirect observations. *IEEE Transactions on reliability*, 59(2):426–439, 2010.

[HA13] K. Hasegawa and Z. I. Awal. A concept for expert system based accident prediction technique for ship maneuvering. In *Proceedings of 5th International Maritime Conference on Design for Safety (IDFS)*, 2013.

[Har76] S. A. Harvald. Factors affecting the stopping ability of vessels. *Journal of International Shipbuilding Progress*, 23(260):106–121, 1976.

[Haw80] D. M. Hawkins. *Identification of outliers*. Springer, 1980.

[Hay98] C. Hayashi. What is data science? fundamental concepts and a heuristic example. In *Data Science, Classification, and Related Methods*, 1998.

[HCS06] G.-B. Huang, L. Chen, and C. K. Siew. Universal approximation using incremental constructive feedforward networks with random hidden nodes. *IEEE Transactions on Neural Networks*, 17(4):879–892, 2006.

[HHSY15] G. Huang, G.-B. Huang, S. Song, and K. You. Trends in extreme learning machines: A review. *Neural Networks*, 61:32–48, 2015.

[HLCS08] G.-B. Huang, M. B. Li, L. Chen, and C. K. Siew. Incremental extreme learning machine with fully complex hidden nodes. *Neurocomputing*, 71(4):576–583, 2008.

[Hop05] H. Hoppe. Goal-based standards. *WMU Journal of Maritime Affairs*, 4(2):169–180, 2005.

[HOT06] G. E. Hinton, S. Osindero, and Y. Teh. A fast learning algorithm for deep belief nets. *Neural computation*, 18(7):1527–1554, 2006.

[HS06] G. E. Hinton and R. R. Salakhutdinov. Reducing the dimensionality of data with neural networks. *Science*, 313:504–507, 2006.

[HSM+00] R. H. R. Hahnloser, R. Sarpeshkar, M. A. Mahowald, R. J. Douglas, and H. S. Seung. Digital selection and analogue amplification coexist in a cortex-inspired silicon circuit. *Nature*, 405:947–951, 2000.

[HU14] M. Hardt and J. Ullman. Preventing false discovery in interactive data analysis is hard. In *Annual Symposium on Foundations of Computer Science*, 2014.

[Hua14] G.-B. Huang. An insight into extreme learning machines: random neurons, random features and kernels. *Cognitive Computation*, 6(3):376–390, 2014.

[Hua15] G.-B. Huang. What are extreme learning machines? filling the gap between frank rosenblatt's dream and john von neumann's puzzle. *Cognitive Computation*, 7(3):263–278, 2015.

[HWB62] J. B. Hadler, C. J. Wilson, and A. L. Beal. Ship standardisation trial performance and correlation with model predictions. In *The Society of Naval Architects and Marine Engineers*, 1962.

[HXL+07] R. Huang, L. Xi, X. Li, C. R. Liu, H. Qiu, and J. Lee. Residual life predictions for ball bearings based on self-organizing map and back propagation neural network methods. *Mechanical Systems and Signal Processing*, 21:193–207, 2007.

[HZ94] G. E. Hinton and R. S. Zemel. Autoencoders, minimum description length and helmholtz free energy. In *Advances in neural information processing systems*, 1994.

[HZS04] G.-B. Huang, Q.-Y. Zhu, and C.-K. Siew. Extreme learning machine: a new learning scheme of feedforward neural networks. In *IEEE International Joint Conference on Neural Networks*, 2004.

[HZS06] G.-B. Huang, Q.-Y. Zhu, and C.-K. Siew. Extreme learning machine: Theory and applications. *Neurocomputing*, 70(1):489–501, 2006.

[ICBR09] F. Immovilli, M. Cocconcelli, A. Bellini, and R. Rubini. Detection of generalized-roughness bearing fault by spectral-kurtosis energy of vibration or current signals. *IEEE Transactions on Industrial Electronics*, 56:4710–4717, 2009.

[IMO12] IMO. *Resolution MSC.337(91), Code on Noise Levels on Board Ships*. London, 2012.

[Int94] International Maritime Organization. Resolution MSC.36(63), International Code of Safety for High-Speed Craft, 1994.

[Int00] International Maritime Organization. Resolution MSC.97(73), International Code of Safety for High-Speed Craft, 2000.

[Int02a] International Maritime Organization. Resolution MSC.137(76), Standards for Ship Maneuverability, 2002.

[Int02b] International Maritime Organization. Resolution MSC/Circ.1053, Explanatory Notes to the Standard for Ship Maneuverability, 2002.

[IPC18] IPCC. Global warming of 1.5°C. Technical report, Intergovernmental Panel on Climate Change, 2018. OCLC: 1056192590.

[ISO04] ISO BS. 13372: 2012, condition monitoring and diagnostics of machines-vocabulary. In *British Standards*, 2004.

[ISO15] Ships and marine technology - guidelines for the assessment of speed and power performance by analysis of speed trial data. Standard, International Organization for Standardization, Geneva, CH, April 2015.

[ISO16a] Ships and marine technology Measurement of changes in hull and propeller performance - Part 1: General principles. Standard, International Organization for Standardization, Geneva, CH, November 2016.

[ISO16b] Ships and marine technology Measurement of changes in hull and propeller performance - Part 2: Default method. Standard, International Organization for Standardization, Geneva, CH, November 2016.

[ISO16c] Ships and marine technology Measurement of changes in hull and propeller performance - Part 3: Alternative methods. Standard, International Organization for Standardization, Geneva, CH, November 2016.

[ISO16d] Mechanical vibration – measurement and evaluation of machine vibration – part 1: General guidelines. Standard, International Organization for Standardization, Geneva, CH, 2016.

[ITT17] ITTC Specialist Committee on Hydrodynamic Noise. Model-scale propeller cavitation noise measurements. Recommended Procedures and Guidelines 7.5-02-01-05, International Towing Tank Conference, 2017.

[JBW+01] A. K. S. Jardine, D. Banjevic, M. Wiseman, S. Buck, and T. Joseph. Optimizing a mine haul truck wheel motors? condition monitoring program use of proportional hazards modeling. *Journal of quality in Maintenance Engineering*, 7(4):286–302, 2001.

[JLB06] A. K. S. Jardine, D. Lin, and D. Banjevic. A review on machinery diagnostics and prognostics implementing condition-based maintenance. *Mechanical systems and signal processing*, 20:1483–1510, 2006.

[K+95] R. Kohavi et al. A study of cross-validation and bootstrap for accuracy estimation and model selection. In *International Joint Conference on Artificial Intelligence*, 1995.

[KBK18] Ž. Koboević, D. Bebić, and Ž. Kurtela. New approach to monitoring hull condition of ships as objective for selecting optimal docking period. *Ships and Offshore Structures*, 0(0):1–9, 2018.

[KCMS16] S. Karmakar, S. Chattopadhyay, M. Mitra, and S. Sengupta. *Induction Motor Fault Diagnosis: Approach Through Current Signature Analysis*. Springer, 2016.

[Kem37] G. Kempf. On the effect of roughness on the resistance of ships. *Trans INA*, 79:109–119, 1937.

[KH07] R. Kothamasu and S. H. Huang. Adaptive mamdani fuzzy model for condition-based maintenance. *Fuzzy Sets and Systems*, 158(24):2715–2733, 2007.

[KL03] S. S. Keerthi and C. J. Lin. Asymptotic behaviors of support vector machines with gaussian kernel. *Neural computation*, 15(7):1667–1689, 2003.

[KOK03] C. T. Kowalski and T. Orlowska-Kowalska. Neural networks application for induction motor faults diagnosis. *Mathematics and Computers in Simulation*, 63:435–448, 2003.

[KS90] G. B. Kliman and J. Stein. Induction motor fault detection via passive current monitoring-a brief survey. In *Meeting of the Mechanical Failures Prevention Group*, 1990.

[KS92] G. B. Kliman and J. Stein. Methods of motor current signature analysis. *Electric Machines and power systems*, 20:463–474, 1992.

[KX11] Y. S. Khor and Q. Xiao. Cfd simulations of the effects of fouling and antifouling. *Ocean Engineering*, 38(10):1065–1079, 2011.

[KZHV13] L. L. C. Kasun, H. Zhou, G.-B. Huang, and C. M. Vong. Representational learning with elms for big data. *IEEE Intelligent Systems*, 28(6):31–34, 2013.

[LB+95] Y. LeCun, Y. Bengio, et al. Convolutional networks for images, speech, and time series. *The handbook of brain theory and neural networks*, 3361(10):1995, 1995.

[LBBH98] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86:2278–2324, 1998.

[LCTH00] B. Li, M. Y. Chow, Y. Tipsuwan, and J. C. Hung. Neural-network-based motor rolling bearing fault diagnosis. *IEEE transactions on industrial electronics*, 47:1060–1069, 2000.

[LDMT16] I. Lazakis, K. Dikis, A. L. Michala, and G. Theotokatos. Advanced ship systems condition monitoring for enhanced inspection, maintenance and decision making in ship operations. *Transportation Research Procedia*, 14:1679–1688, 2016.

[LLST13] G. Lever, F. Laviolette, and F. Shawe-Taylor. Tighter pac-bayes bounds through distribution-dependent priors. *Theoretical Computer Science*, 473:4–28, 2013.

[LLYB13] G. Langxiong, L. Liangming, Z. Yuangzhou, and Z. Baogang. A new method for accurate prediction of ship's inertial stopping distance. *Research Journal of Applied Sciences, Engineering and Technology*, 18(6):3437–3440, 2013.

[LM07] H. Liu and H. Motoda. *Computational methods of feature selection*. CRC Press, 2007.

[LMJR17] M. Lützen, L.L. Mikkelsen, S. Jensen, and H.B. Rasmussen. Energy efficiency of working vessels – A framework. *Journal of Cleaner Production*, 143:90–99, 2017.

[Log12] K. P. Logan. Using a ship's propeller for hull condition monitoring. *Naval Engineers Journal*, 124(1):71–87, 2012.

[LSSV08] L. Leifsson, H. Sævarsdóttir, S. Sigurdhsson, and A. Vésteinsson. Grey-box modeling of an ocean vessel for operational optimization. *Simulation Modelling Practice and Theory*, 16(8):923–932, 2008.

[LW02] A. Liaw and M. Wiener. Classification and regression by randomforest. *R News*, 2(3):18–22, 2002.

[MA97] B. Maines and R.E.A. Arndt. The case of the singing vortex. *Journal of fluids engineering*, 119(2):271–276, 1997.

[McC62] B. W. McCormick. On cavitation produced by a vortex trailing from a lifting surface. *Journal of Basic Engineering*, 84(3):369–378, 1962.

[MCI84] Great Britain. Quality Management, Statistics Standards Committee, and British Standards Institution. *British Standard Glossary of Maintenance Management Terms in Terotechnology*. British Standards Institution, 1984.

[MCO+19] F. Miglianti, F. Cipollini, L. Oneto, G. Tani, and M. Viviani. Model scale cavitation noise spectra modelling: Combining physical knowledge with data science. *Ocean Engineering*, 2019.

[MD03] S. A. McInerny and Y. Dai. Basic vibration signal processing for bearing fault detection. *IEEE Transactions on education*, 46:149–156, 2003.

[MEP18] MEPC. Meeting Summary of the Marine Environment Protection Committee (MEPC), 72nd Session. Technical Report, Maritime Environmental Protection Committee (MEPC), part of the International Maritime Organisation (IMO), London, United Kingdom, 2018.

[MLY13] L. Ming, J. X. Liu, and S. Yang. A new method on calculation of vessels stopping distance and crash stopping distance. In *Advanced Materials Research*, 2013.

[MMO18] M. Mirovic, M. Milicevic, and I. Obradovic. Big data in the maritime industry. *Nase More*, 65, 2018.

[Mob02] R. K. Mobley. *An introduction to predictive maintenance*. Butterworth-Heinemann, 2002.

[MS03] M. Markou and S. Singh. Novelty detection: a review. *Signal processing*, 83(12):2481–2497, 2003.

[MS10] A. Mannini and A. M. Sabatini. Machine learning methods for classifying human physical activity from on-body accelerometers. *Sensors*, 10(2):1154–1175, 2010.

[MSK95] L. Mann, A. Saxena, and G. M. Knapp. Statistical-based or condition-based preventive maintenance? *Journal of Quality in Maintenance Engineering*, 1(1):46–59, 1995.

[MW15] N. Marz and J. Warren. *Big Data: Principles and best practices of scalable realtime data systems*. Manning Publications Co., 2015.

[NCL+11] J. Ngiam, A. Coates, A. Lahiri, B. Prochnow, Q. V. Le, and A. Y. Ng. On optimization methods for deep learning. In *Proceedings of the 28th international conference on machine learning (ICML-11)*, 2011.

[Nik13] G. Nikolakaki. Economic incentives for maritime shipping relating to climate protection. *WMU Journal of Maritime Affairs*, 12(1):17–39, 2013.

[NTL05] S. Nandi, H. A Toliyat, and X. Li. Condition monitoring and fault diagnosis of electrical motors—a review. *IEEE transactions on energy conversion*, 20:719–729, 2005.

[OCC+18] L. Oneto, A. Coraddu, F. Cipollini, O. Karpenko, Sanetti P. Xepapa, K., and D. Anguita. Crash stop maneuvering performance prediction: a data driven solution for safety and collision avoidance. *Data-Enabled Discovery and Applications*, 2(1):11, 2018.

[ODO+18] D. Owen, Y. K. Demirel, E. Oguz, T. Tezdogan, and A. Incecik. Investigating the effect of biofouling on propeller characteristics using cfd. *Ocean Engineering*, 159:505 – 516, 2018.

[ÖDS06] I. Y. Önel, K. B. Dalci, and I. Senol. Detection of bearing defects in three-phase induction motors using park's transform and radial basis function neural networks. *Sadhana*, 31, 2006.

[OFC+17] L. Oneto, E. Fumeo, C. Clerico, R. Canepa, F. Papa, C. Dambra, N. Mazzino, and Anguita. D. Dynamic delay predictions for large-scale railway networks: Deep and shallow extreme learning machines tuned via thresholdout. *IEEE Transactions on Systems, Man and Cybernetics: Systems*, 47(10):2754 – 2767, 2017.

[One18] L. Oneto. Model selection and error estimation without the agonizing pain. *WIREs Data Mining and Knowledge Discovery*, 2018.

[OOA16] I. Orlandi, L. Oneto, and D. Anguita. Random forests model selection. In *ESANN*, 2016.

[OTNS74] H. Okamoto, A. Tanaka, K. Nozawa, and Y. Saito. Stopping abilities of vessels equipped with controllable pitch propeller. *Journal of International Shipbuilding Progress*, 21(235):53–69, 1974.

[PB07] M. J. Pazzani and D. Billsus. Content-based recommendation systems. In *The adaptive web*, 2007.

[PBA+11] T. Palmé, P. Breuhaus, M. Assadi, A. Klein, and M. Kim. New alstom monitoring tools leveraging artificial neural network technologies. In *Turbo Expo: Turbine Technical Conference and Exposition*, 2011.

[PCE+13] M. D. Prieto, G. Cirrincione, A. G. Espinosa, J. A. Ortega, and H. Henao. Bearing fault detection by a novel condition-monitoring scheme based on statistical-time features and neural networks. *IEEE Transactions on Industrial Electronics*, 60:3398–3407, 2013.

[Pea01] Karl Pearson. Principal components analysis. *The London, Edinburgh and Dublin Philosophical Magazine and Journal*, 6:566, 1901.

[PJH03] S. Pöyhönen, P. Jover, and H. Hyötyniemi. Independent component analysis of vibrations for fault diagnosis of an induction motor. In *Proceedings of the International Conference Circuits, Signals, and Systems (IASTED)*, 2003.

[PJW12] J. P. Petersen, D. J. Jacobsen, and O. Winther. Statistical modelling for ship propulsion efficiency. *Journal of marine science and technology*, 17(1):30–39, 2012.

[PK14] H. N. Psaraftis and C. A. Kontovas. Ship speed optimization: Concepts, models and combined speed-routing scenarios. *Transportation Research Part C: Emerging Technologies*, 44:52–69, 2014.

[PL09] B. P. Pedersen and J. Larsen. Prediction of full-scale propulsion power using artificial neural networks. In *The 8th International Conference Computer and IT Applications in the Maritime Industries*, 2009.

[Rae96] A. E. Raestad. Tip vortex index-an engineering approach to propeller noise prediction. *The Naval Architect*, pages 11–15, 1996.

[Ras06] C. E. Rasmussen. Gaussian processes for machine learning. In *Gaussian Processes for Machine Learning*, 2006.

[RHW88] D. E. Rumelhart, G. E. Hinton, and R. J. Williams. Learning representations by back-propagating errors. *Cognitive modeling*, 5:1, 1988.

[Ros58] F. Rosenblatt. The perceptron: a probabilistic model for information storage and organization in the brain. *Psychological review*, 65:386, 1958.

[RRS00] S. Ramaswamy, R. Rastogi, and K. Shim. Efficient algorithms for mining outliers from large data sets. In *ACM Sigmod Record*, 2000.

[RRZ97] S. Ridella, S. Rovetta, and R. Zunino. Circular backpropagation networks for classification. *IEEE Transactions on Neural Networks*, 8(1):84–97, 1997.

[RV15] A. Radonjic and K. Vukadinovic. Application of ensemble neural networks to prediction of towboat shaft power. *Journal of Marine Science and Technology*, 20(1):64–80, 2015.

[RYP03] R. Rifkin, G. Yeo, and T. Poggio. Regularized least-squares classification. *Nato Science Series Sub Series III Computer and Systems Sciences*, 190:131–154, 2003.

[S+03] G. K. Singh et al. Induction machine drive condition monitoring and diagnostic research—a survey. *Electric Power Systems Research*, 64:145–158, 2003.

[SA11] J. T. Selvik and T. Aven. A framework for reliability and risk centered maintenance. *Reliability Engineering & System Safety*, 96(2):324–331, 2011.

[SABAA03] B. Samanta, K. R. Al-Balushi, and S. A. Al-Araimi. Artificial neural networks and support vector machines with genetic algorithm for bearing fault detection. *Engineering Applications of Artificial Intelligence*, 16:657–665, 2003.

[SAG16] K. C. Seo, M. Atlar, and B. Goo. A study on the hydrodynamic effect of biofouling on marine propeller. *Journal of The Korean Society of Marine Environment & Safety*, 22(1):123–128, 2016.

[SC07] H. Su and K. T. Chong. Induction machine condition monitoring using neural network modeling. *IEEE Transactions on Industrial Electronics*, 54:241–249, 2007.

[Sch90] R.L. Schiltz. Forcing frequency identification of rolling element bearings. *Sound and vibration*, 24(5):16–19, 1990.

[Sch01] B. Scholkopf. The kernel trick for distances. In *Advances in neural information processing systems*, 2001.

[Sch04] M. P. Schultz. Frictional resistance of antifouling coating systems. *Journal of fluids engineering*, 126(6):1039–1047, 2004.

[Sch07] M. P. Schultz. Effects of coating roughness and biofouling on ship resistance and powering. *Biofouling*, 23(5):331–341, 2007.

[Sch15] J. Schmidhuber. Deep learning in neural networks: An overview. *Neural networks*, 61:85–117, 2015.

[SGJ09] Y. T. Shen, S. Gowing, and S. Jessup. Tip vortex cavitation inception scaling for high reynolds number applications. *Journal of Fluids Engineering, American Society of Mechanical Engineers*, 131(7), 2009.

[SGK95] R. M. Singer, K. C. Gross, and R. W. King. A pattern-recognition-based, fault-tolerant monitoring and diagnostic technique. In *Argonne National Lab., Idaho Falls, ID (United States)*, 1995.

[SH97] R. R. Schoen and T. G. Habetler. Evaluation and implementation of a system to eliminate arbitrary load effects in current-based monitoring of induction machines. *IEEE Transactions on Industry Applications*, 33:1571–1577, 1997.

[SHH04] J. R. Stack, T. G. Habetler, and R. G. Harley. Fault classification and fault signature production for rolling element bearings in electric machines. *IEEE Transactions on Industry Applications*, 40(3):735–739, 2004.

[SHHB12] J. Schröder-Hinrichs, E. Hollnagel, and M. Baldauf. From titanic to costa concordia—a century of lessons not learned. *WMU Journal of Maritime Affairs*, 11(2):151–167, 2012.

[SHK+14] N. Srivastava, G. E Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *Journal of machine learning research*, 15:1929–1958, 2014.

[SHKB95] R. R. Schoen, T. G. Habetler, F. Kamran, and R. G. Bartfield. Motor bearing damage detection using stator current monitoring. *IEEE transactions on industry applications*, 31:1274–1279, 1995.

[SHS01] B. Schölkopf, R. Herbrich, and A. J. Smola. A generalized representer theorem. In *Computational learning theory*, 2001.

[SJA+14] T. W. P. Smith, J. P. Jalkanen, B. A. Anderson, J. J. Corbett, J. Faber, S. Hanayama, E. O'Keeffe, S. Parker, L. Johansson, L. Aldous, et al. Third international maritime organization green house gas study. *International Maritime Organization (IMO), London*, 327, 2014.

[SLH+95] R. R. Schoen, B. K. Lin, T. G. Habetler, J. H. Schlag, and S. Farag. An unsupervised, on-line system for induction motor fault detection using stator current monitoring. *IEEE Transactions on Industry Applications*, 31:1280–1286, 1995.

[SLK05] K.-S. Shin, T. S. Lee, and H.-J. Kim. An application of support vector machines in bankruptcy prediction model. *Expert Systems with Applications*, 28(1):127–135, 2005.

[Smi44] N. V. Smirnov. Approximate laws of distribution of random variables from empirical data. *Uspekhi Matematicheskikh Nauk*, 10:179–206, 1944.

[SMR07] V. Sugumaran, V. Muralidharan, and K. I. Ramachandran. Feature selection using decision tree and classification through proximal support vector machine for fault diagnostics of roller bearing. *Mechanical systems and signal processing*, 21:930–942, 2007.

[SMS+16] L. Swersky, H. O. Marques, J. Sander, R. J. G. B. Campello, and A. Zimek. On the evaluation of outlier detection and one-class classification methods. In *IEEE International Conference on Data Science and Advanced Analytics*, 2016.

[SSBD14] S. Shalev-Shwartz and S. Ben-David. *Understanding machine learning: From theory to algorithms*. Cambridge university press, 2014.

[SSÇU14] H. Saruhan, S. Sandemir, A. Çiçek, and I. Uygur. Vibration analysis of rolling element bearings defects. *Journal of applied research and technology*, 12(3):384–395, 2014.

[SSE03] M. L. Sin, W. L. Soong, and N. Ertugrul. Induction machine on-line condition monitoring and fault diagnosis-a survey. In *Australasian Universities Power Engineering Conference*, 2003.

[STC04] J. Shawe-Taylor and N. Cristianini. *Kernel methods for pattern analysis*. Cambridge university press, 2004.

[Ste37] E. A. Stevens. The increase in frictional resistance due to the action of water on bottom paint. *Naval Engineers Journal*, 49(4):585–588, 1937.

[SvdLGP18] W. Schim van der Loeff, J. Godar, and V. Prakash. A spatially explicit data-driven approach to calculating commodity-specific shipping emissions per vessel. *Journal of Cleaner Production*, 205:895–908, 2018.

[TA79] A. N. Tikhonov and V. Y. Arsenin. *Methods for solving ill-posed problems*. Nauka, Moscow, 1979.

[TAVA17] G. Tani, B. Aktas, M. Viviani, and M. Atlar. Two medium size cavitation tunnel hydro-acoustic benchmark experiment comparisons as part of a round robin test campaign. *Ocean Engineering*, 138:179–207, 2017.

[TDH16] J. Tang, C. Deng, and G. Huang. Extreme learning machine for multilayer perceptron. *IEEE transactions on neural networks and learning systems*, 27(4):809–821, 2016.

[TF01] W. T. Thomson and M. Fenger. Current signature analysis to detect induction motor faults. *IEEE Industry Applications Magazine*, 7:26–34, 2001.

[Tib96] R. Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, 58(1):267–288, 1996.

[TM16] M. D. Tissera and M. D. McDonnell. Deep extreme learning machines: supervised autoencoding architecture for classification. *Neurocomputing*, 174:42–49, 2016.

[Tsa02] A. H. C. Tsang. Strategic dimensions of maintenance management. *Journal of Quality in Maintenance Engineering*, 8:7–39, 2002.

[TVAN16] G. Tani, M. Viviani, E. Armelloni, and M. Nataletti. Cavitation tunnel acoustic characterisation and application to model propeller radiated noise measurements at different functioning conditions. *Proceedings of the Institution of Mechanical Engineers, Part M: Journal of Engineering for the Maritime Environment*, 230(2):250–266, 2016.

[Vap98] V. N. Vapnik. *Statistical learning theory*. Wiley New York, 1998.

[Vas93] P. Vas. *Parameter estimation, condition monitoring, and diagnosis of electrical machines*. Oxford: Clarendon Press; New York: Oxford University Press, 1993.

[VLBM08] P. Vincent, H. Larochelle, Y. Bengio, and P. A. Manzagol. Extracting and composing robust features with denoising autoencoders. In *Proceedings of the 25th international conference on Machine learning*, 2008.

[WCGT09] Z. X. Wang, C. S. Chang, X. German, and W. W. Tan. Online fault detection of induction motors using independent component analysis and fuzzy neural network. In *International Conference on Advances in Power System Control, Operation and Management*, 2009.

[Whi00] H. White. A reality check for data snooping. *Econometrica*, 68:1097–1126, 2000.

[Wir12] D. I. F. Wirz. Optimisation of the crash-stop manoeuvre of vessels employing slow-speed two-stroke engines and fixed pitch propellers. *Journal of Marine Engineering & Technology*, 11(1):35–43, 2012.

[WSF15] H. Winnes, L. Styhre, and E. Fridell. Reducing ghg emissions from ships in port areas. *Research in Transportation Business & Management*, 17:73–82, 2015.

[WY07] A. Widodo and B. Yang. Support vector machine in machine condition monitoring and fault diagnosis. *Mechanical systems and signal processing*, 21(6):2560–2574, 2007.

[YKDJ04] D. M. Yebra, S. Kiil, and K. Dam-Johansen. Antifouling technology - past, present and future steps towards efficient and environmentally friendly antifouling coatings. *Progress in Organic Coatings*, 50(2):75–104, 2004.

[YKP+97] B. Yazici, G. B. Kliman, W. J. Premerlani, R. A. Koegl, G. B. Robinson, and A. Abdel-Malek. An adaptive, on-line, statistical method for bearing fault detection using stator current. In *Industry Applications Conference, 1997. Thirty-Second IAS Annual Meeting, IAS'97., Conference Record of the 1997 IEEE*, 1997.

[You03] D. M. Young. *Iterative solution of large linear systems*. DoverPublications, 2003.

[ZH05] H. Zou and T. Hastie. Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 67(2):301–320, 2005.

[ZHL+15] H. Zhou, G. B. Huang, Z. Lin, H. Wang, and Y. C. Soh. Stacked extreme learning machines. *IEEE Transactions on Cybernetics*, 45(9):2013–2025, 2015.

[ZHSK04] A. Zeileis, K. Hornik, A. Smola, and A. Karatzoglou. kernlab-an s4 package for kernel methods in r. *Journal of statistical software*, 11(9):1–20, 2004.

[ZLHH09] W. Zhou, B. Lu, T. G. Habetler, and R. G. Harley. Incipient bearing fault detection via motor stator current noise cancellation using wiener filter. *IEEE Transactions on Industry Applications*, 45:1309–1317, 2009.

[ZS10] Y.P. Zhao and J.G. Sun. Robust truncated support vector regression. *Expert Systems with Applications*, 37(7):5126–5133, 2010.